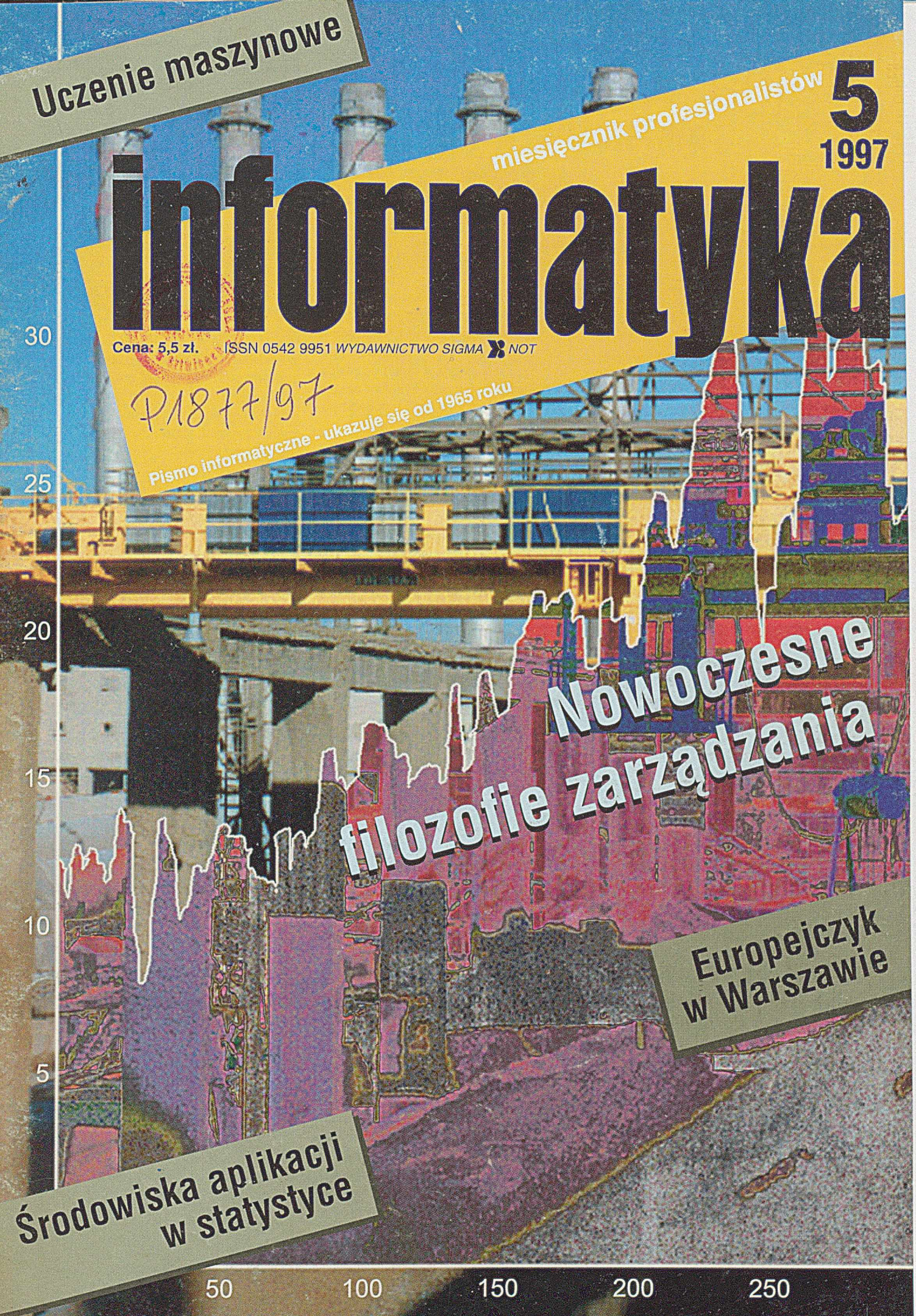
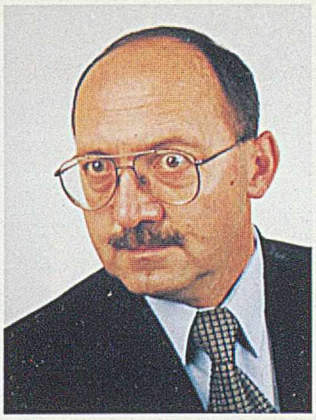

Uczenie maszynowe
miesięcznik profesjonalistów
5
1997
informatyka
Cena: 5,5 zł.
ISSN 0542 9951 WYDAWNICTWO SIGMA NOT
P1877/97
Pismo informatyczne - ukazuje się od 1965 roku
Nowoczesne filozofie zarządzania
Europejczyk w Warszawie
Środowiska aplikacji w statystyce

Tajna wizyta
emisariusza integracji



Kwietniowa wizyta w Polsce dr. Martina Bangemanna wywołała spore poruszenie w środowiskach związanych z teleinformatyką. Komisarz Unii Europejskiej (odpowiednik ministra w rządzie krajowym) ds. przemysłu, technologii informacyjnych i telekomunikacji, znany jest przede wszystkim jako koordynator prac nad raportem o globalnym społeczeństwie informacyjnym, zwanym też Raportem Bangemanna.

Przebieg wizyty, dla tych którzy mieli możność ją „zza płotu" obserwować, jak i sposób informowania o niej mogły zaszokować. Tak ważne – wydawałoby się – wydarzenie prawie umknęło uwadze mediów. Nie powiadomiono o nim Polskiej Agencji Prasowej, a z kilku spotkań jakie odbył w Polsce dr Bangemann nie ukazał się ani jeden komunikat prasowy. W dniu wizyty w Wiadomościach (dawniej Dziennik Telewizyjny) znalazł się czas na reportaż o zerwanym przez wichurę dachu, za to nie zamieszczono relacji z rozmów jakie prowadził Komisarz Unii. Można się tylko domyślić, że dotyczyły one przyszłości teleinformatyki w Polsce.

Wysoki urzędnik Unii wypraszał ze swoich spotkań dziennikarzy, a dopiero potem rozmawiał z przedstawicielami administracji, szefami wybranych przedsiębiorstw, stowarzyszeń i organizacji branżowych. Można sądzić, że nie życzył sobie by polskie społeczeństwo znało przebieg rozmów. A przecież dyskutowano (?) o newralgicznym i najbardziej dochodowym sektorze gospodarki, który jest w rozwiniętych krajach – zależnie od punktu widzenia – ekonomicznym koniem pociągowym lub dojną krową. Dr Bangemann, wieloletni szef niemieckiej partii wolnych demokratów (FDP), jest oczywiście apostołem liberalizmu, a więc rozwiązań wolnorynkowych w gospodarce. Pozostanie jego tajemnicą, jak pogodzić wolny rynek ze sterowanym dostępem do ważnych informacji.

Dotychczas w okresie przystosowywania gospodarki każdego z krajów kandydujących do UE prowadzono zawsze rzeczowy i otwarty dialog o ewentualnych stratach i zyskach wynikających z tego faktu dla firm, branż i sektorów. Według niektórych opinii negocjacje z Unią są sprawą zbyt delikatną, aby publikować ich szczegóły zanim nie zapadną choćby wstępne ustalenia. Inni zaś uważają, że otaczanie tajemnicą wizyt ważnych polityków, w tym przypadku z Unii, sprzyja różnym spekulacjom i rodzi obawy. W każdym kraju nie brakuje tzw. eurosceptyków, którzy zechcą je wykorzystać. Polacy zaś, szczególnie kiedy nie identyfikują się ze strukturą polityczną i gospodarczą, w której się znajdą, nie są dobrym sojusznikiem. Aby nas pozyskać trzeba nam zaufać i traktować poważnie.

Proces wchodzenia do Unii, jak też obecność w niej, są kosztowne dla obu stron, tak przyjmowanych, jak i przyjmujących, a dyskusje i negocjacje dotyczą podziału tych kosztów. Stosunkowo proste jest dopasowanie przepisów prawa i norm, zaś bardzo poważne mogą się okazać skutki zmian: praw regulujących funkcjonowanie gospodarki, ceł, barier ochronnych, cen czy przepisów podatkowych. W przypadku Polski całe dziedziny gospodarki, jak np. rolnictwo, przemysł paliwowy, hutnictwo, będą miały kłopoty z przetrwaniem w obecnej postaci.

Kłopoty te ominą (niestety) przemysł elektroniczny, bo został już zlikwidowany. Mamy za to około 6 tys. różnej wielkości firm związanych z teleinformatyką, zatrudniających tysiące specjalistów. Powinni oni wiedzieć jakie są plany Unii względem polskiego sektora teleinformatycznego, aby mieli szansę uniknąć losu wrocławskiego ELWRO, czy warszawskiego CEMI. Tym, którzy twierdzą, że ten sektor sam się obroni bo jest w większości prywatny przypomnijmy, że rolnictwo też jest prywatne.

Telekomunikacja w Polsce to głównie Telekomunikacja Polska S.A. Jest to przedsiębiorstwo o największym w Polsce dochodzie brutto (ponad 500 mln USD), zatrudniające prawie 75 tys. ludzi. W wyniku uzgodnień międzynarodowych będzie w najbliższych latach prywatyzowane i zostaną mu odebrane atrybuty monopolisty. Komisarz Bangemann nie spotkał się podczas swojej wizyty w Polsce z przedstawicielami TP S.A. Odwołał też bez merytorycznej przyczyny zaplanowaną wizytę w Komitecie Badań Naukowych, choć to jego departament decyduje w Unii o środkach przyznawanych na badania w zakresie telekomunikacji i informatyki. Podsumowując, wizyta dr. Bangemanna nie była udana od strony propagandowej. Czy powinniśmy jednak utożsamiać niekonwencjonalny styl Komisarza, z jakiego jest znany, z polityką „15" ?

Wejście do Unii spowoduje, oprócz przemian gospodarczych, także wprowadzenie europejskich standardów dziennikarstwa, które polegają na informowaniu o prawdziwych i ważnych problemach. Konflikty interesów, zwłaszcza gospodarczych, są w Unii na porządku dziennym, ale nie są ukrywane czy nie zauważane, lecz rozwiązywane. Społeczności, których one dotyczą, wiedzą z czym strony przystępują do negocjacji i co uzgodniono.

W powodzi aplikowanych nam ciągle haseł: Unia Europejska, NATO, integracja, infostrady, multimedia, demonopolizacja, prywatyzacja itp., zaczynamy gubić sens tych terminów. Na przykład informatyka ma swój źródłosłów w informacji, a telekomunikacja – w komunikacji. Bez „komunikowania informacji" takie urządzenia jak telefony, komputery, kable, modemy, satelity są tylko górą żelastwa i plastiku, tak jak bezużytecznym meblem jest komputer bez systemu operacyjnego. Społeczeństwo informacyjne jest nie tylko „okablowane", ale także, a może przede wszystkim – rzetelnie poinformowane. W ub.r. na X Światowym Kongresie „Technologie i Usługi w Społeczeństwie Informacyjnym" dr Bangemann powiedział: „*Technologie informatyczne są potężnym narzędziem, ale to tylko środek do osiągnięcia naszych ekonomicznych i społecznych celów*". Otóż to, Panie Komisarzu.

Jerzy Dworzecki

informatyka

Miesięcznik
ISSN 0542-9951
INDEKS 36124

POLITECHNIKA ŚLĄSKA Biblioteka Główna w Gliwicach
P1877/97

Ukazuje się elektroniczna wersja naszego miesięcznika. Oto nasz adres w Internecie: **www.pol.pl/informatyka/**

nr 5 1997
Maj
Rok wydania XXXIII

REDAGUJE ZESPÓŁ:
dr Lesław **WAWRZONEK**
(redaktor naczelny)
mgr Jerzy **SZYLLER**
(redaktor)
mgr Tomasz **SZEWCZYK**
(redaktor)
mgr Zdzisław **ŻURAKOWSKI**
(redaktor)
mgr inż. Agnieszka **ZIÓŁKOWSKA**
(sekretarz redakcji)
Alina **Klepacz**
(redakcja techniczna, administracja)

KOLEGIUM REDAKCYJNE:
prof.dr hab. Leonard BOLC
mgr inż. Piotr FUGLEWICZ
prof. dr hab. Jan GOLIŃSKI
dr inż. Zenon KULPA
prof. dr inż. Jan MULAWKA
prof. dr hab. Wojciech OLEJNICZAK
mgr. inż. Jan RYŻKO
dr Witold STANISZKIS
dr inż. Jacek STOCHLAK
prof. dr hab. Maciej STOLARSKI
prof. dr hab. Zdzisław SZYJEWSKI
prof. dr hab. inż.
Ryszard TADEUSIEWICZ
prof. dr hab. Jan WĘGLARZ

PRZEWODNICZĄCY
RADY PROGRAMOWEJ
prof. dr hab. Juliusz Lech KULIKOWSKI

WYDAWCA:
Wydawnictwo Czasopism i Książek Technicznych SIGMA NOT Spółka z o.o.
ul. Ratuszowa 11
00-950 WARSZAWA
skrytka pocztowa 1004

REDAKCJA:
00-950 Warszawa,
ul. Ratuszowa 11, p. 628
skrytka pocztowa 1004
tel., fax: 619-11-61
tel.: 619-22-41 w. 159
e-mail: informat@pol.pl

PRENUMERATA:
tel. 40-35-86, 40-30-86

Materiałów nie zamówionych redakcja nie zwraca.

Autorzy artykułów proszeni są o przysyłanie tekstów na dyskietkach 3½" - w czystych kodach ASCII (Latin II, Mazovia) lub edytorach: Word 2, Word 6.
Po szczegółowe informacje dla Autorów prosimy zwracać się do redakcji.

Redakcja nie ingeruje w treść i formę ogłoszeń i innych materiałów reklamowych, w związku z tym nie ponosi za nie odpowiedzialności.

Ogłoszenia przyjmują:
- Redakcja, tel. 619-11-61
- Dział Reklamy i Marketingu
00-950 Warszawa, ul. Mazowiecka 12
tel.: 827-43-66, fax: 826-80-16

Okładka:
AGAT, Jerzy Burski i Andrzej Jacyszyn,
Łamanie:
Alina Klepacz, program PageMaker 5.0
Druk:
Drukarnia SIGMA NOT Sp. z o.o.

W numerze:

Strona

INFORMACJE 2

SYSTEMY INFORMACYJNE

Business Process Reengineering na tle nowoczesnych koncepcji zarządzania 6
Marek Tokarski

Spójna sieć komputerowa przedsiębiorstwa, cz. II 11
Zdzisław Wrzeszcz

ANALIZY

Polskie oprogramowanie teraz i w przyszłości 19
Janusz Gieruszyński

KONFERENCJE

Doświadczenia w zakresie zastosowania metryk do oceny jakości oprogramowania 22
Janusz Górski, Marek Jurkowlaniec, Bartosz Nowicki

KOMPUTER OSOBISTY

Konsekwencja w każdym calu 25
Po prostu WinTel 28
Tomasz Szewczyk

PUBLIKACJE

Uczenie maszynowe – obiecującą strategią przetwarzania informacji w biznesie, cz. II 29
Zdzisław S. Hippe

Systemy czasu rzeczywistego

Programowanie systemów czasu rzeczywistego w Adzie 95 34
Zbigniew Fryźlewicz, Zbigniew Huzar

BIULETYN PTI 42

Apel o wsparcie funduszu

Grono przyjaciół **Marka Cara**, dziennikarza i animatora polskiego internetu, postanowiło utworzyć fundusz, który wykorzystany będzie w przyszłości na cele edukacji Jego córek Joanny (13 lat) i Katarzyny (2 lata).

Zwracamy się z prośbą o wsparcie tego celu. Zebrane środki przeznaczone zostaną na zakup polis ubezpieczeniowych o charakterze inwestycyjnym, z których dochody pomogą pokrywać koszty kształcenia.

Fundusze na zakup polis gromadzimy na koncie:

BIG S.A. I/Oddział Warszawa
Kopernika 36/40
00-924 Warszawa
r-k: 11601029-02803008
Polska OnLine - MAREK CAR

Zip zamiast napędów 3,5"

Niebawem pojawi się na rynku nowy produkt firmy **Iomega** - Zip wyposażony w interfejs ATAPI. Firma chce doprowadzić do zastąpienia napędów dyskietek 3.5 cala dużo bardziej pojemnym Zip'em. Producenci układów BIOS wprowadzili już do najnowszych wersji możliwość startowania systemu z oprogramowania zapisanego na Zip'ie, a w niedalekiej przyszłości powinny ukazać się na rynku płyty główne wyposażone w BIOS z tą opcją.

Microsoft i Novell zapomnieli o języku polskim

Novell zawarł umowę z **Microsoft Corp.** dotyczącą wykorzystania w oprogramowaniach Microsoftu mechanizmów językowych opracowanych przez Novella. Oprogramowanie obejmuje Novell Collexion Grammar Checker - narzędzie pozwalające na sprawdzenie pisowni i poprawności gramatycznej dokumentów napisanych w wielu językach (francuski, holenderski, niemiecki i hiszpański) oraz Novell Collexion Morphology - narzędzie lingwistyczne umożliwiające odnalezienie rdzeni wyrazów, wspomagające sprawdzanie gramatyki i zawierające funkcje przeszukiwania.

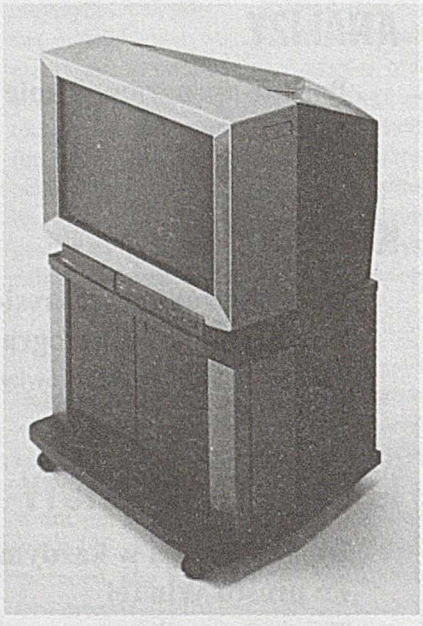
Oprogramowanie do zarządzania szpitalem

Firma APEXIM podjęła się lokalizacji (tłumaczenie i adaptacja do lokalnych przepisów) pakietu oprogramowania AMIS*H czeskiej firmy AMIS (Advanced Medical Information Systems). W celu realizacji zadania została założona spółka **APEXIM & AMIS-POLSKA**. Pierwsze moduły zostaną przekazane za około dwa miesiące.

✍ Są więc szanse na to, że zarządzanie polskimi szpitalami wyjdzie poza narzędzia pakietu Office.

Nowoczesny kineskop

Kineskop **Super Flat Trinitron** firmy Sony został zastosowany w telewizorze KV-28SF5. Ekran w kineskopie Trinitron jest wycinkiem powierzchni walca. Nowy układ stabilizujący oraz maskownica ze szczeliną 0,56 mm zapewniają bardzo wysoką rozdzielczość obrazu. W kineskopie zastosowano zmodyfikowaną wyrzutnię elektronową o ogniskowej 35 mm i kącie odchylenia



zwiększonym do 102°, dającą równomierne ogniskowanie obrazu na całej powierzchni ekranu. Dzięki precyzyjnemu układowi odchylenia z cewkami o zwiększonej średnicy zminimalizowano rozszczepienie kolorów oraz zniekształcenia geometrii obrazu w narożnikach ekranu. Ekran kineskopu pokryty jest wielowarstwową powłoką antyodblaskową. Dzięki funkcji Twin Picture można oglądać dwa programy jednocześnie, indeks z podziałem ekranu na 12 pól pozwala szybko wybrać żądany kanał.

Kontrolery SCSI dla każdego

California Computer podpisała umowę dystrybucyjną z DPT, specjalizującym się w technologii SCSI. **Kontrolery SCSI serii SmartCacheIV** mają konstrukcję modularną, umożliwiającą rozbudowę kontrolera o moduł zapewniający sprzętowy cache i sprzętową implementację RAID. Ponadto można je rozbudować o dodatkowe szyny SCSI (maksymalnie 3 szyny z jednego kontrolera). Kontrolery DPT zyskały uznanie dzięki wersjom współpracującym z kilkoma typami magistral (PCI, ISA, EISA) i wieloma systemami operacyjnymi, jak np.: Novell Netware, IBM OS/2 2.1 i OS/2 Warp, MS Windows (3.1, 95, NT), SunSoft Solaris, SCO Unix, NeXT NEXTSTEP, BSD i BSD/OS oraz Linux.

Komputerowe próby z bronią jądrową

Rosyjskie

Ośrodek badań nad bronią jądrową w Rosji **zakupił przez pośrednika w Europie**, przetwarzający równolegle komputer IBM RS/6000 SP, omijając w ten sposób przepisy eksportowe Stanów Zjednoczonych. Rosjanie poinformowali, że planują go wykorzystać do symulowania prób jądrowych. Rząd amerykański twierdzi, że superkomputery mogłyby pomóc Rosjanom w zaprojektowaniu nowej broni. Zgodnie z kodeksem prawnym USA amerykańscy producenci komputerów muszą otrzymać zgodę Waszyngtonu na sprzedaż swoich najwydajniejszych komputerów.

✍ To chyba dobrze, że także w Rosji będzie mniej skażeń radioaktywnych.

Amerykańskie

IBM i LLNL (Lawrence Livermore Laboratory) zawarły kontrakt na wykonanie programu badawczo - rozwojowego Departamentu Energii - ASCI Blue (Accelerated Supercomputing Initiative).

W ramach programu ma być zbudowany do końca 1998 roku system komputerowy o maksymalnej wydajności 3,2 mld operacji zmiennoprzecinkowych na sekundę (3,2 GFLOP). ASCI jest progamem rozwoju zaawansowanych symulacji komuterowych, które mogą zastąpić rzeczywiste eksperymenty jądrowe z użyciem prawdziwej broni.

✍ Działacze Green Peace będą musieli się przenieść z malowniczego archipelagu Mururoa do szacownego Berkeley w Kaliforni, aby protestować przciw próbom z bronią jądrową w miejscu gdzie są wykonywane. **(lw)**

XIII Konferencja INFORMATYKA W SZKOLE

W dniach 17-20 września 1997 r. odbędzie się w Lublinie na Uniwersytecie Marii Curie-Skłodowskiej XIII Konferencja *Informatyka w Szkole*. Przewodnim tematem konferencji będzie ***„Integrująca rola technologii informacyjnej"***. W programie są przewidziane również: wykłady plenarne, obrady w sesjach tematycznych nt. nauczanie przedmiotu: elementy informatyki, diagnozowania osiągnięć uczniów z informatyki, miejsca komputerów w pedagogice i szkolnictwie specjalnym, stosowania komputerów w nauczaniu matematyki, fizyki, przedmiotów przyrodniczych i przedmiotów zawodowych, kształcenie w zawodzie technik informatyk, od internetu do intranetu.

Informacje można uzyskać w Instytucie Informatyki Uniwersytetu Wrocławskiego, ul. Przesmyckiego 20, 51-151 Wrocław; tel. (0-71) 325 12 71.

Monitor z kineskopem Trinitron i z głośnikami

Ukazały się monitory 449Xa i 449Xi firmy **Nokia**. Model 449Xa to 15 calowy monitor z kineskopem Trinitron, z obrazem 278 x 207 mm i maksymalnej rozdzielczości 1024 x 768 przy 80 Hz (800 x 600 przy 100 Hz). Monitor ma wbudowane głośniki 2 x 2 W. Oba modele mają certyfikaty ergonomiczne i certyfikaty bezpieczeństwa. Monitory są kompatybilne z systemem Macintosh i Plug-&-Pley.

Nowy Map Publisher

Map Publisher firmy **Intergraph** został przystosowany do pracy z Windows NT i Windows 95. Oprogramowanie jest środowiskiem do obróbki rastrowych map wielkoformatowych. Dane wektorowe są konwertowane do obrazu rastrowego o wysokiej rozdzielczości, który po obróbce gotowy jest do druku offsetowego. Map Publisher ma funkcje: zarządzanie projektem, rendering kartograficzny, prepress control oraz wiele opcji druku.

Samokształcenie szefów dużych przedsięwzięć informatycznych

Kierujący dużymi przedsięwzięciami informatycznymi spotkali się w kwietniu Warszawie na Warsztatach Menedżerskich organizowanych przez **Andrzeja Florczyka** - Dyrektora Biura Informatyki w Krajowym Biurze Wyborczym. Nie pierwsze już takie spotkanie, było tym razem poświęcone podsumowaniu dokonań i planów na przyszłość grupy ludzi, zainteresowanych szerzeniem kultury menedżerskiej i podnoszeniem kwalifikacji w zakresie zarządzania dużymi projektami informatycznymi. Według zaproponowanego przez Andrzeja Florczyka kryterium, duże projekty to przedsięwzięcia o wartości przekraczającej 1 mln USD. Dodał, że około 500 menedżerów w Polsce jest odpowiedzialnych za 120 takich projektów, których koszt przekracza rocznie 3 mld USD.

Celem Warsztatów Menedżerskich jest, zdaniem ich inicjatora, stworzenie bazy kadrowej dla dużych przedsięwzięć informatycznych. Działalność organizacyjna polega na stworzeniu warunków do wymiany doświadczeń, także z szefami firm, organizacji i instytucji, którzy są lub będą odbiorcami takich projektów. Wiadomo, dodał Andrzej Florczyk, że za szybkim rozwojem technologii nie nadąża, nie tylko w Polsce, postęp organizacyjny i liczba doświadczonych kadr. Innym z zadań takiego forum mogłoby być, jego zdaniem, określenie przejrzystych kryteriów wyboru technologii do realizacji projektów informatycznych, aby nie dawać podstaw do podejrzeń o korupcję.

Organizator spotkania stwierdził też, że niepochlebna społeczna opinia o dużych przedsięwzięciach informatycznych została ukształtowana przez media, które interesują się tylko wpadkami - „Bardzo udany projekt jakim jest informatyzacja NFI został przemilczany przez prasę".

Wiesław Paluszyński - Dyrektor Departamentu Bezpieczeństwa Wewnętrznego Biura Bezpieczeństwa Narodowego przy Kancelarii Prezydenta uważa, że poważną przeszkodą w realizacji udanych projektów jest brak zainteresowania niektórych odbiorców sukcesem takich przedsięwzięć. Wynika to z konieczności uporządkowania organizacji instytucji, której projekt dotyczy, co niektórym tam zatrudnionym może zaszkodzić - „Dobre zarządzanie projektem i prawidłowe jego wykonanie nie zawsze wystarczą do stworzenia sprawnie działającego systemu". Dodał, że brakuje specjalistów, którzy mogliby poprowadzić duże przedsięwzięcia informatyczne dla administracji i samorządów.

Zbigniew Chwaliński Dyrektor Biura Informatyki Komendy Głównej Policji mówił m.in. o trudnościach z dotarciem do informacji o najnowszych technologiach, co przy szybkim postępie powoduje, że projekt po zakończeniu może okazać się przestarzały. Zdarza się - mówił - że krajowe przedstawicielstwa firm wiedzą mniej o ofercie ich centrali niż prasa. Dotyczy to także szczególnie dla niego ważnych zabezpieczeń dostępu do sieci.

Inicjatywę powołania i kontynuacji działań Warsztatów Menedżerskich poparł **Lech Szukszta** - Dyrektor Departamentu Informatyki w Narodowym Banku Polskim. Za ich zaletę uznał możliwość bezpośredniego kontaktu kierujących projektami informatycznymi z ich użytkownikami i potencjalnymi odbiorcami. „Ludzie z forsą widzą świat zupełnie inaczej i warto ich punkt widzenia poznać" - stwierdził. Warsztaty, to jego zdaniem, nowoczesna organizacja samoucząca, której nie zastąpi w tworzeniu kadry menedżerskiej żadna uczelnia czy kursy.

Czy firmy o dużym doświadczeniu menedżerskim mogą kształcić taką kadrę? **Małgorzata Kalinowska-Iszkowska** Dyrektor Ośrodka Szkolenia IBM Polska, powiedziała, że choć IBM ma duże jest doświadczenie w zarządzaniu, to jednak niechętnie dzieli się swoją wiedzą prowadząc kursy w tej dziedzinie. Firma InfoVide natomiast, specjalizująca się w szkoleniach - jak powiedział jej dyrektor **Borys Stokalski** - nie może zaspokoić dużego i rosnącego zapotrzebowania na kadry menedżerskie.

Bez entuzjazmu do pomysłu Andrzeja Florczyka, formalnego stowarzyszenia firm szkoleniowych w celu określenia jakiegoś standardu szkoleń menedżerskich odniósł się **Maciej Wierzba**, prezes InfoVide. Stwierdził, że kontakty towarzyskie szefów takich firm na razie zupełnie wystarczają.

Tematmi kolejnych Warsztatów Menedżerskich będą: informatyzacja banków i zasady podejmowania decyzji wyboru technologii w przedsięwzięciach informatycznych.

Lesław Wawrzonek

(*Wypowiedzi rozmówców nie są autoryzowane*)

V Międzynarodowe Targi Zastosowań Informatyki w Przemyśle i CAD/CAM-97

W dniach 10-13 czerwca 1997 r. w Pałacu Kultury i Nauki w Warszawie odbędą się targi organizowane przez Stowarzyszenie Użytkowników Systemów Inżynierskich (SUSI) i Biuro Reklamy SA Zarządu Targów Warszawskich oraz Agencję Techniki i Technologii. Patronat nad targami objęło Ministerstwo Skarbu Państwa. Targom towarzyszyć będzie cykl prezentacji, seminariów i konferencji, nad czym czuwać będzie specjalnie powołana Rada Programowa, w skład której wejdą przedstawiciele Ministerstwa Skarbu Państwa, instytucji naukowych i przemysłu.

Amerykańskie analizy polskiego rynku

Od 15 lutego br. firma **Hogart reprezentuje GartnerGroup**, jedną z bardziej renomowanych na świecie firmę specjalizującą się w działalności doradczej oraz analizowaniu, opiniowaniu i interpretowaniu istotnych postępów i zmian w dziedzinie informatyki.

Polskie firmy na konkurencyjnych rynkach

Wiele polskich firm eksportuje własne programy i usługi programistyczne na trudne rynki europejskie i amerykańskie. Stowarzyszenie Polski Rynek Oprogramowania postanowiło podkreślić sukcesy polskich firm software'owych na rynkach światowych. Podczas Konferencji „Polski Rynek Oprogramowania 96/97" Zarząd PRO przyznał tytuł „Lidera Eksportu Oprogramowania '96" firmie **Young Digital Poland** z Gdańska za skuteczne wprowadzenie na konkurencyjne rynki europejskie polskiego oprogramowania edukacyjnego. Wyróżnienia otrzymały także firmy **Arcus Electronics** z Wrocławia i **Inter-Design Tessel** Systems z Warszawy.

Porozumienie CSBI i ORACLE Polska

CSBI S.A. i ORACLE Polska podpisały porozumienie o współpracy. Dotyczyć ono będzie przede wszystkim baz danych ORACLE i rozszerzy ofertę CSBI o te produkty. CSBI zyskuje z jednej strony poszerzenie swojej oferty handlowej w zakresie baz danych, a z drugiej zwiększa możliwości wdrażania i rozbudowywania swych aplikacji tworzonych narzędziami środowiska PROGRESS również dla tych klientów, którzy dysponują już lub zechcą opierać swe bazy danych na produktach ORACLE.

Usługi informatyczne dla banków w Polsce

ComputerLand Poland SA i Sanchez Computer Associates podpisały umowę o współpracy w zakresie usług informatycznych dla banków w Polsce. Obie firmy będą mieć wspólną strategię marketingową, usług i sprzedaży do polskich banków.

Polskie drukarki kodów kreskowych

Zakłady **Mera Błonie** rozpoczęły produkcję **drukarek kodów kreskowych - AP 105**. Jest to polski wyrób, w którym wykorzystano technologię francuskiej firmy New Code. Dzięki zastosowaniu technologii termotransferu użytkownicy otrzymają wysokiej jakości samoprzylepne etykiety.

Złoty Medal Targów Infosystem '97

OKIFAX 4100, najnowszy faks firmy Oki został laureatem Złotego Medalu Targów Infosystem '97. Okifax 4100 to nie tylko faks, ale także drukarka, modem, skaner i kopiarka. Rozdzielczość druku wynosi 600 dpi. Zajmuje niewiele miejsca (szerokość 312 mm, głębokość 383 mm, wysokość190 mm).

Scenic Pro M5, jest komputerem PC najnowszej generacji z czytnikiem kart chipowych.

W tym komputerze produkcji Siemens Nixdorf został zainstalowany czytnik kart chipowych jako opcja pod nazwą SiCrypt. Nagrodzony produkt, wraz z towarzyszącym mu oprogramowaniem stanowi doskonałe zabezpieczenie sprzętowe przed niepowołanym dostępem do danych zgromadzonych w systemie.

Informatyka w przemyśle

CSBI S.A. zawarła z DAEWOO-FSO MOTOR Sp. z o.o. kontrakt na wdrożenie systemu MFG/PRO oraz dostawę i wdrożenie współdziałających z tym systemem modułów Kadry, Płace, Środki Trwałe, Wyposażenie, RCP, Przelewy Elektroniczne systemu PRO/MIS. Wdrożenie zostanie przeprowadzone w Zakładzie DAEWOO-FSO Żerań w Warszawie oraz w Zakładzie Samochodów Dostawczych w Nysie. W ramach kontraktu CSBI dostarczy także oprogramowanie narzędziowe PROGRESS. Wartość kontraktu **wynosi ok. 1,0** mln. USD.

Pol-Mot Praszka będzie wrdażać system informatyczny wspomagający zarządzanie przedsiębiorstwem oferowany przez **IFS Poland**. Umowa obejmuje następujące moduły: finanse, dystrybucja, produkcja, remonty. Platformą sprzętową będzie serwer IBM RS/6000 G40 obsługujący 96 użytkowników. Dostawca zapewnia także szkolenia, doradztwo, instalację sprzętu oraz opiekę powdrożeniową. Zakończenie prac planowane jest na czerwiec 1999 r.

Serwery dla internetu i intranetu

Sun i Komtech oferują kompletne systemy serwerów do tworzenia aplikacji rozproszonych dla środowiska internet oraz intranet. Serwer SuMag to platforma środowiska Solaris'a i Magic'a skonfigurowana do projektowania i wdrażania w sieci internet lub intranet dynamicznego serwisu WWW działającego w oparciu o dowolną bazę danych.

Oferta składa się z następujących części:

- serwera Netra j;
- kompletnego środowiska systemowego Solaris;
- platforma do tworzenia aplikacji języku JAVA;
- środowiska generatora aplikacji relacyjnych baz danych MAGIC v.7.

Serwery Netra to komputery do realizacji bezpiecznego i wydajnego połączenia sieci lokalnej z siecią internet. Serwery te są dostarczane z niezbędnym oprogramowaniem, w tym z pakietem FireWall-First. Serwery Netra można wykorzystać również w konfiguracji intranetu do świadczenia serwisu WWW użytkownikom sieci lokalnej.

Magic Web został zaprojektowany dla klientów, którzy potrzebują narzędzia do tworzenia skalowalnych, zorientowanych na transakcyjne przetwarzanie, dynamicznych stron WWW. Pozwala na tworzenie aplikacji internetowych współpracujących z kilkoma serwerami Magic'a, działającymi na różnych maszynach.

Nabywcom serwerów SUMAG zamawiającym jednocześnie stacje robocze firmy Sun Microsystems oferuje 35%

zniżkę na te stacje zamiast standardowej 30% zniżki akademickiej.

Promocja SUN i Magic dla wyższych uczelni

W kwietniu br. SUN Microsystems Polska i Komtech Sp. z o.o. Radom ogłosiły wspólną promocję dla Uczelni zestawów sprzętowo-software'owych przeznaczonych do tworzenia aplikacji w internecie. Główne elementy zestawu to: komputer Netra J z pakietem oprogramowania (w tym Java i Fire Wall) oraz pakiet Magic Dev. V. 7, Magic Serwer na SUN Solaris i Magic Web.

Oprogramowanie Magic dostarczone w ramach promocji umożliwia równoczesne tworzenie aplikacji przez 20-stu programistów.
Promocja obowiązuje w kwietniu i maju br. Szczegółowe informacje można uzyskać w Internecie :
WWW. komtech.com.pl.lub WWW. poland.sun.com

AS/400 komunikuje się z Windows NT

6 kwietnia 1997 podczas konferencji COMMON'97 firmy **Digital Equipment Corporation i Microsoft Corporation** zademonstrowały rozwiązanie umożliwiające użytkownikom systemu AS/400, korzystającym z takich aplikacji jak hurtownie danych, poczta elektroniczna oraz działającym za pośrednictwem Internetu, współpracę ze środowiskiem Windows NT.

Na całość oprogramowania firmy Microsoft, umożliwiającego integrację ze środowiskiem systemów AS/400 składa się wiele modułów takich jak serwer Microsoft BackOffice, oprogramowanie kliencki Microsoft Office, Microsoft SQL Server, wysokowydajny system zarządzania bazą danych, Microsoft Exchange Server oraz Microsoft Systems Management Server, który jest podstawowym narzędziem dla zarządzania komputerami PC w środowisku sieci LAN.

Prezent klasy MRP II dla uczelni

Ster-Projekt podpisał umowę z **Wydziałem Zarządzania Uniwersytetu Warszawskiego**, dzięki której studenci tego wydziału zapoznają się na zajęciach ze zintegrowanym systemem do zarządzania przedsiębiorstwem klasy MRP II. W ramach umowy Ster-Projekt dostarczył komputer, na którym zainstalowany został system do zarządzania Triton, holenderskiej firmy Baan. Specjaliści firmy pomagają wykładowcom Uniwersytetu w opracowaniu odpowiedniego programu zajęć dla studentów, który pozwoli na jeszcze lepsze przygotowanie zawodowe przyszłych menedżerów.

Światowy Internet

Firma **Polska Online** jako pierwsza w Polsce wprowadziła do swojej oferty nową usługę przeznaczoną dla ludzi, którzy chcieliby bez względu na miejsce pobytu mieć możliwość skorzystania z internetu bez konieczności dokony wania międzynarodowych połączeń do Polski. Usługa nazwana PoL Global ROAM umożliwia korzystanie ze wszystkich internetowych usług, w tym WWW i poczty elektronicznej, podczas pobytu w dowolnym ze 159 krajów, w których działają operatorzy internetowi związani umową i-PASS Alliance.

Europejczyk w Warszawie

Z przedstawicielami środowiska polskiego biznesu teleinformatycznego spotkał się 10 kwietnia w Warszawie **dr Martin Bangemann**, Komisarz Unii Europejskiej kierujący XIII Dyrektoriatem Generalnym (DG XIII). Urząd ten, o randze ministerstwa, jest odpowiedzialny za przemysł, technologie informacyjne i telekomunikację.

Na spotkanie, które trwało ok. 2 godzin, przybyli reprezentanci (w większości dyrektorzy i prezesi) największych polskich firm teleinformatycznych: CENTERTEL ComputerLand, CSBI, EDS Poland, GSM Plus, IBM Polska, Microsoft Polska, Optimus, TCH, przedstawiciele PIIiT i stowarzyszeń: PRO, PTI, SRSO, SUSI. Ze względu na wyraźne żądanie Gościa, nie mogli być obecni w czasie trwania dyskusji zaproszeni na spotkanie przedstawiciele prasy: Informatyka, PCKurier, Rzeczpospolita i Teleinfo. Można przypuszczać, że dyskusja dotyczyła demonopolizacji polskiego rynku telekomunikacyjnego i barier ochronnych dla branży teleinformatycznej.

W swoim wystąpieniu wprowadzającym dr Bangemann powiedział, że Unia popiera i wita z sympatią zamiar wstąpienia do niej Polski. Stwierdził, że jest znacznie lepiej dla zainteresowanych wejściem w struktury europejskie, gdy długofalowe cele i polityka gospodarcza danego kraju i Unii są uzgadniane i modernizowane jeszcze przed przyjęciem. Celem jego wizyty w Polsce jest poznanie na miejscu skali przyszłych problemów w tym zakresie.

Stwierdził, że w krajach Europy Zachodniej prywatyzacja rynku telekomunikacyjnego wynikała z konieczności zainwestowania w ten sektor dużych pieniędzy (jak można sądzić, ze względu na konkurencję amerykańską), a tego nie mogły udźwignąć budżety państw - właścicieli krajowych operatorów. Ponadto - dodał - nikt nie będzie inwestował w system monopolistyczny, który może nagle się zmienić. Problemy monopolistów telekomunikacyjnych są tak samo ważne dla kandydatów jak i dla krajów już będących w Unii, w której także nie zostały one do końca rozwiązane. Rynek telekomunikacji - stwierdził - regulowały prawa monopolu, były to jednak prawa. Nowy zdemonopolizowany i sprywatyzowany rynek telekomunikacji również wymaga - jego zdaniem - regulacji, ale innych, nowoczesnych. Uczestnicy tego rynku muszą przestrzegać praw autorskich, standardów, zasad konkurencji. Konieczność nowych regulacji prawnych wynika też z postępu technicznego, charakteryzującego się zacieraniem granic między różnymi typami mediów -mówił dr Bangemann. Na przykład nie da się stosować do nowych technik multimedialnych regulacji prawnych, które były odpowiednie dla tradycyjnej telewizji.

fot. L. Wawrzonek

dr Bangemann (w centrum)

Lesław Wawrzonek

(*Wypowiedzi dr. Bangemanna nie są autoryzowane*).

dr Martin Bangemann

Urodzony w 1934 r w Niemczech. Jest dr. prawa, a z racji wykonywanego przez wiele lat zawodu - politykiem. Był posłem do Bundestagu z ramienia Partii Wolnych Demokratów (FDP), w której zajmował stanowisko sekretarza generalnego, a następnie przewodniczącego. Od 1973 do 1984 w Parlamencie Unii Europejskiej. W latach 1984-88 federalny minister finansów. Od 1989 ponownie na wysokich stanowiska w UE. Koordynator raportu o globalnym społeczeństwie informacyjnym zwanym też Raportem Bangemanna.

Business Process Reengineering
na tle nowoczesnych koncepcji zarządzania

Marek Tokarski

Rozpoczynający się tym artykułem cykl ma na celu zapoznanie czytelników z problematyką szeroko rozumianej koncepcji zarządzania zwanej Business Process Reengineering.

W kolejnych artykułach zamierzam:
1. ukazać tę koncepcję na tle nowoczesnych filozofii zarządzania,
2. wyjaśnić istotę reengineeringu i rolę czynników umożliwiających zmiany oraz dokonać przeglądu metod przeprojektowania,
3. ukazać rolę technologii informatycznych w tej koncepcji,
4. omówić kwestie związane z wdrażaniem reengineeringu,
5. wyciągnąć praktyczne wnioski odnośnie stosowania tej techniki przez polskie przedsiębiorstwa.

W polskim piśmiennictwie koncepcję tę rozważa się często jedynie w kontekście wdrażania technologii informatycznych, jednak filozofia ta jest o wiele bardziej złożona i posiada liczne inne aspekty. Uważam, że nowoczesny menedżer lub konsultant zajmujący się stosowaniem technologii informatycznych powinien poznać szersze tło tej koncepcji, aby lepiej poznać jej istotę i skutecznie stosować ją w praktyce gospodarczej.

Tło tej koncepcji stanowi nowoczesny paradygmat zarządzania, który wymaga zastosowania nowej filozofii i narzędzi zarządzania. Te narzędzia zarządzania powstały w wyniku niekończącej się pogoni mającej na celu uzyskanie wyższej produktywności, jakości i szybkości produkcji oraz operacji współczesnych przedsiębiorstw. Wśród tych koncepcji profesor Harvard Business School Michael E. Porter wymienia: kompleksowe zarządzanie jakością (total quality management), benchmarking[1], konkurowanie czasem (time-based competition), outsourcing[2], budowanie partnerstwa firm (partnering) oraz zarządzanie zmianami (change management)[1]. Aby zrozumieć miejsce Business Process Reengineeringu niezbędne jest poznanie genezy nowoczesnych koncepcji zarządzania.

[1] Proces porównywania efektów działalności z najlepszymi w swej kategorii (a także innych branż) przedsiębiorstwami i organizacjami. Celem tego procesu jest określenie sposobu w jaki najlepsi osiągają tak znakomite efekty, czyli poszukiwanie najlepszych praktyk.
Źródło: Grzegorz M. Augustyniak: Rola Kompleksowego Zarządzania Jakością (TQM) w strategii konkurencyjnej przedsiębiorstw, niepublikowana praca badawcza, Katedra Logistyki SGH, 1995, s .76.

[2] Koncepcja postulująca zakup usług u zewnętrznych oferentów.

Rozwój nowoczesnych koncepcji zarządzania

Do lat pięćdziesiątych nie istniała alternatywa dla produkcji masowej. Większość przedsiębiorstw produkowała w sposób, który wypracowany został przez takie autorytety zarządzania jak: F. W. Taylor, H. Ford czy A. Sloan. Jednak stopniowo wraz ze wzrostem nasycenia rynku produktami, rosły oczekiwania klientów i coraz trudniej było sprzedać wytworzony produkt.

W latach sześćdziesiątych doszło do tzw. rewolucji konsumenckiej, oczekiwania klientów znacznie wzrosły, rozpoczęło się badanie potrzeb i oczekiwań klienta w celu dostarczenia satysfakcjonujących je produktów. W zachodnich firmach dostrzeżono wagę marketingu. Zarządy firm przestały koncentrować się na problemach produkcji i zaczęto zwracać uwagę na problemy związane ze sprzedażą.

Natomiast w dziedzinie produkcji starano się jedynie utrzymać status quo, wynikające ze światowych standardów efektywności. Inaczej oceniali zaistniałą sytuację menedżerowie japońscy. Menedżerowie firmy Toyota zauważyli niewydolność zachodnich systemów produkcji masowej i próbowali określić jej przyczyny.

Okazało się, iż system masowej produkcji posiada kilka istotnych wad:

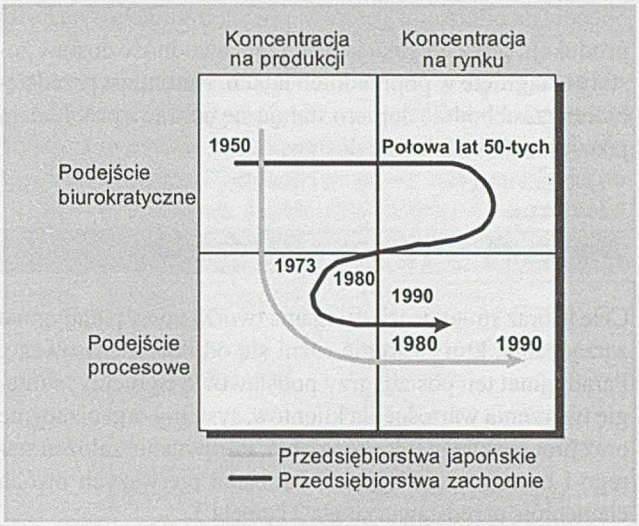
1. Produkcja charakteryzowała się niską jakością i konieczne było usuwanie wielkiej liczby usterek, występujących w gotowych wyrobach tuż po zejściu z linii produkcyjnej.
2. W celu nadzoru słabo zmotywowanych pracowników niezbędny był kosztowny system funkcjonalnej, hierarchicznej kontroli.
3. Produkcja nie była dostosowana do zmieniających się potrzeb rynku, a przestawienie na produkcję nowego modelu zatrzymywało ją na długie miesiące i powodowało dalszy wzrost kosztów.
4. Utrzymywano nadmierne ilości zapasów materiałów, komponentów i wyrobów gotowych.

W latach pięćdziesiątych powstawał system produkcyjny firmy Toyota, który sprawił, że cały świat zarządzania zaczął patrzeć na Japonię. Eiji Toyoda i Taiichi Ohno rozpoczęli tworzenie systemu zarządzania, który w latach osiemdziesiątych doprowadził do sukcesu gospodarki japońskiej i globalnej ekspansji przedsiębiorstw japońskich, które przejęły metody zarządzania Toyoty.

Dla gospodarki amerykańskiej przełomowym rokiem był rok 1973, rok kryzysu naftowego, kiedy to przedsiębiorstwa amerykańskie okazały się kompletnie nieprzystosowane do zmian rynkowych. Po tym roku w Stanach Zjednoczonych, a następnie w Europie pojawiły się takie firmy jak: Toyota, Nissan i Honda tworząc nowe standardy efektywności działania nieosiągalne dla przedsiębiorstw amerykańskich i europejskich.

Wtedy to dopiero praktycy i teoretycy zarządzania zaczęli badać japońskie systemy zarządzania. Na ich podstawie stworzono w latach osiemdziesiątych i dziewięćdziesiątych wiele nowoczesnych koncepcji zarządzania takich jak: zarządzanie dokładnie na czas (just-in-time), kompleksowe zarządzanie jakością (total quality management), zarządzanie wyszczuplające (lean management), inżynieria równoległa (simultaneous engineering) oraz zarządzanie uwzględniające czynnik czasu (time-based management).

Filozofie te opierają się na podobnych, podstawowych założeniach, ich porównanie przedstawiono w Tabeli 1.



Rys. 1. Orientacja procesowa i rynkowa przedsiębiorstw japońskich i zachodnich.
Źródło: Georgio Merli: Total Manufacturing Management: Production Organization for the 1990s, Productivity Press 1990,

Koncepcja zarządzania Podstawowe założenia	Kompleksowe zarządzanie jakością (TQM)	Just- In-Time (JIT)	Inżynieria równoległa (SE)	Time-based management (TBM)	Lean management (LM)
Kluczowa rola klienta	Tak	Nie	Nie	Nie	Tak
Poprawa płynności przepływów	Nie	Tak	Tak	Tak	Tak
Stałe doskonalenie	Tak	Tak	Nie	Nie	Tak
Pełne zaangażowaniu pracowników i praca zespołowa	Tak	Tak	Nie	Nie	Tak
Optymalizacja utrzymywanych zapasów	Nie	Tak	Nie	Nie	Tak
Uelastycznienie systemu produkcji	Nie	Tak	Tak	Tak	Tak
Eliminacja czynności nie tworzących wartości oraz marnotrawstwa	Tak	Tak	Tak	Tak	Tak
Kompresja czasu cyklu	Nie	Tak	Tak	Tak	Nie

Źródło: Opracowanie własne.

Tabela 1. Porównanie podstawowych założeń nowoczesnych koncepcji zarządzania.

Wymienione filozofie mają różnie rozłożone akcenty. Podkreślają np. ciągłe doskonalenie i perfekcyjną jakość (TQM), czynnik czasu (TBM) lub na płynność procesu i obniżenie wielkości utrzymywanych zapasów (JIT). Koncepcje te są próbą wyjaśnienia fenomenu przedsiębiorstw japońskich i przetłumaczenia niektórych japońskich technik na zachodni sposób rozumowania.

Przewaga firm japońskich

Pomimo prób wdrażania tych koncepcji zarządzania, firmy zazwyczaj nie osiągają trwałej przewagi. Z czego to wynika? Przede wszystkim z faktu, że firmy zachodnie używają japońskich metod zarządzania w celach taktycznych, a nie strategicznych. Ponadto przedsiębiorstwa japońskie posiadają wieloletnie doświadczenia w tej dziedzinie podczas gdy ich zachodni konkurenci dopiero się uczą.

Japończycy już od 1973 roku rozwijają podejście procesowe włączające zewnętrznych kooperantów w swoje systemy produkcyjne, tworząc łańcuch dostaw. Ponadto od początku lat osiemdziesiątych ich produkcja jest ściśle powiązana z oczekiwaniami konsumentów. Przedsiębiorstwa zachodnie zaczęły stosować koncepcję łańcucha dostaw na początku lat osiemdziesiątych (pierwsze zachodnie próby stosowania systemu Just-In-Time). Natomiast orientacja rynkowa w produkcji pojawiła się na początku lat dziewięćdziesiątych, czego wyrazem było pełne wdrożenie metody projektowania sterowanego przez klienta (quality function deployment) i nowoczesnych, elastycznych systemów wytwarzania (flexible manufacturing system).

Firmy zachodnie są opóźnione w upowszechnieniu orientacji rynkowej i stosowaniu podejścia procesowego średnio o około 10 lat w porównaniu z firmami japońskimi. W większości polskich przedsiębiorstw podobne opóźnienie występuje w stosunku do przedsiębiorstw zachodnich. Zależności te i opóźnienie firm zachodnich w stosunku do firm japońskich przedstawiono na rysunku 1.

To dziesięcioletnie opóźnienie wynika z faktu, że od połowy lat pięćdziesiątych menedżerowie zachodni koncentrowali się głównie na doskonaleniu produktów oraz ich marketingu i sprzedaży, podczas gdy japończycy rozpoczęli doskonalenie procesów produkcyjnych. Dzięki temu w latach osiemdziesiątych możliwe stało się wbudowanie „perspektywy klienta" w proces projektowania produktu. Stworzyło to nową jakość konkurowania na rynkach globalnych. W latach osiemdziesiątych na zachodzie dostrzeżono konsekwencje tego faktu i zaczęto studiować i przejmować metody przedsiębiorstw japońskich, co nie wyeliminowało jednak powstałego opóźnienia.

Różnica 10 lat wynika również ze ściśle ustalonej hierarchii celów japońskich przedsiębiorstw, którymi są:

- wysoka jakość,
- niezawodność dostaw,
- niższe koszty produkcji,
- zwiększona elastyczność produkcji [2].

Zazwyczaj są one realizowane w podanej sekwencji. Większość japońskich przedsiębiorstw koncentruje się

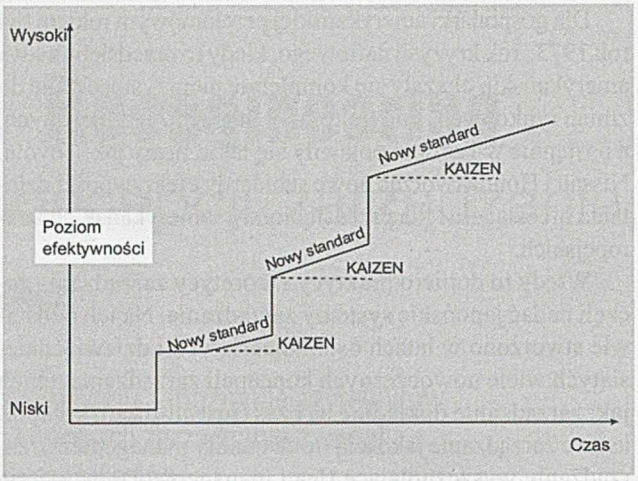
obecnie na obniżaniu kosztów i zwiększaniu elastyczności produkcji, gdyż wysoka jakość i niezawodność dostaw zostały osiągnięte w poprzednich latach. Natomiast przedsiębiorstwa zachodnie dopiero starają się uporać z problemem jakości i niezawodności dostaw.

Nowy paradygmat zarządzania

Cele te oraz strategie ich osiągania tworzą nowy paradygmat zarządzania, który istotnie różni się od dotychczasowego. Paradygmat ten posiada trzy podstawowe elementy: strategię tworzenia wartości dla klientów, systemy organizacyjne oraz proces stałego udoskonalania. Porównanie założeń starego i nowego paradygmatu odnośnie pierwszych dwóch elementów przedstawia tabela 2 i tabela 3.

Trzecim elementem jest ciągłe doskonalenie, które wyrosło z japońskiej filozofii kaizen. Kaizen jest filozofią zarządzania, która w sposób kompleksowy obejmuje wszystkie obszary działalności gospodarczej oraz koncentruje się na procesie drobnych, stopniowych ale częstych usprawnień. Termin ten opisuje więc dążenie organizacji do stopniowej, stałej i ciągłej poprawy operacyjnej efektywności.

Tradycyjne innowacje po początkowym osiągnięciu wyższej efektywności mają tendencję spadkową. Natomiast innowacje wspomagane przez filozofię kaizen zazwyczaj stabilizują osiągnięty poziom efektywności, a nawet wywołują drobny jej wzrost. Zależność tę przedstawia rysunek 2.



Rys. 2. Kaizen a innowacje
Źródło: M. Imai: Kaizen: The Key to Japanese Competitive Sucess, Random House 1986, New York, p.27.

W teorii i praktyce zarządzania szeroko dyskutuje się istotę ciągłego doskonalenia. Kwestię tą rozważa kompleksowe zarządzanie jakością. Według klasycznych teorii opisujących innowacje wyróżnia się jej następujące formy:

- innowacje radykalne albo innowacje stopniowe,
- innowacje techniczne lub innowacje kierownicze oraz
- innowacje produktowe i procesowe [3].

Innowacje produktowe były jednocześnie innowacjami technicznymi oraz radykalnymi, a innowacje procesowe - kierowniczymi i stopniowymi. Teoretycy zarządzania wielokrotnie zastanawiali się nad wewnętrznymi i zewnętrznymi źródłami innowacji oraz nad tworzeniem organizacji, która byłaby nastawiona na wyszukiwanie warunków sprzyjających innowacjom i ich wdrażaniu [4].

Zagadnienie	Tradycyjny paradygmat	Nowy paradygmat
Jakość	Zgodność ze standardami, inspekcja produktu, współzależność pomiędzy jakością, kosztem i planami	Jeden komponent - „wartość dla klienta", jakość zarządzana w procesie, poszukiwanie synergii pomiędzy jakością, kosztem i planami
Miary sprawności	Wewnętrzne mierniki wydajności, produktywności, kosztów i rentowności, niekoniecznie powiązane z klientami	Wszystkie mierniki powiązane z „wartością" tworzoną dla klienta
Plasowanie	Konkurencyjne	Segmenty rynku
Kluczowy udziałowiec	Udziałowcy, szef, pozostali udziałowcy są nieistotni	Klient, pozostali udziałowcy są beneficjentami
Projektowanie produktu	Wewnętrzne, sprzedaż produktów, które można wyprodukować	Zewnętrzne, produkcja spełniająca potrzeby klienta

Tabela 2. Porównanie starego i nowego (kształtującego) się paradygmatu zarządzania - strategia tworzenia wartości dla klienta.
Źródło: Greg Bounds, Lyle Yorks, Mel Adams, Gipsie Ranney: Beyond Total Quality Management. Toward the Emerging Paradigm, McGraw-Hill 1994, New York, p. 29.

Zagadnienie	Tradycyjny paradygmat	Nowy paradygmat
Podejście interfunkcjonalne	Negocjacje na styku pionów funkcjonalnych, w celu osiągnięcia współpracy	Zdefiniowane, zoptymalizowane systemy interfunkcjonalne
Technologia	Pokonywanie złożoności, eliminacja problemów osobowych	Obniżenie złożoności, źródło optymalizacji w celu maksymalizacji wartości tworzonej dla klienta
Włączanie pracowników	Skupione na czynnikach operacyjnych	Skupione na czynnikach strategicznych
Zarządzanie zasobami ludzkimi	Rutynowa czynność, zarządzanie personelem, zatrudnianie i zwalnianie oraz załatwianie skarg	Zasób krytyczny dla sukcesu firmy, zarządzanie jako zasób na wejściu firmy
Definiowanie ról	Zadania i zakres obowiązków wyznaczają obszar działania	Wizja inspiruje elastyczność

Tabela 3. Porównanie starego i nowego (kształtującego)się paradygmatu zarządzania - systemy organizacyjne
Źródło: Greg Bounds, Lyle Yorks, Mel Adams, Gipsie Ranney: Beyond Total Quality Management. Toward the Emerging Paradigm, McGraw-Hill 1994, New York, p. 31.

Geneza Business Process Reengineeringu

Jednak jak udowadniają Michael Hammer oraz Thomas H. Davenport i James E. Short mogą istnieć także inne, odmienne od klasycznych źródła skokowego wzrostu efektywności organizacji. Takimi przyczynami mogą być radykalne zmiany organizacyjne, wpierane przez podejście procesowe i innowacyjne stosowanie technologii komputerowych[5]. Próbując opisać takie podejście klasycznymi kategoriami innowacji można je sklasyfikować jako innowacje procesowe, kierownicze i równocześnie techniczne, o charakterze radykalnym.

Jeżeli połączymy to twierdzenie z sytuacją, która zaistniała w gospodarce światowej w latach dziewięćdziesiątych, w których przedsiębiorstwa muszą podołać wzrostowi natężenia konkurencji oraz postępującemu procesowi globalizacji, to można zauważyć przyczyny, dzięki którym technika zarządzania stwarzająca perspektywy błyskawicznego nadrobienia dystansu do liderów rynkowych stała się tak atrakcyjna dla menedżerów na całym świecie.

Szczególnie widoczne jest to w Stanach Zjednoczonych, gdzie powstała ta koncepcja. Business Process Reengineering był przedstawiany przez swoich twórców jako rdzennie amerykańska technika, która była odpowiedzią na zalew japońskich koncepcji zarządzania. Był to „rdzennie amerykański przejaw przedsiębiorczości". Oprócz narodowego charakteru obiecywano menedżerom zmniejszenie dystansu do japońskich konkurentów, którzy od lat doskonalili się w uzyskiwaniu operacyjnej efektywności [6].

Tak więc Business Process Reengineering jest programem jednorazowym, który pozwala osiągnąć istotny postęp w podstawowych miarach sprawności i efektywności organizacji. W opinii Michaela Hammera oraz Jamesa Champy'ego reengineering „nie ma prawie nic wspólnego z dotychczasowymi filozofiami zarządzania, koncentrującymi się na ciągłym doskonaleniu i odróżnia się w istotny sposób nawet od tych podejść, z którymi dzieli pewne założenia" [7].

Według nich reengineering jest „nowym początkiem", który wymaga „odrzucenia konwencjonalnej wiedzy i przejętych z przeszłości założeń ..." [7] oraz „odkryciem nowych podejść do struktur procesowych, które mają niewiele wpólnego z tymi z poprzednich er."[7].

Natomiast Thomas H. Davenport i James E. Short uważają, że reengineering jest naturalnym rozwinięciem dotychczasowych filozofii zarządzania, wywodzących się z koncepcji naukowego zarządzania, zapoczątkowanych przez Taylora [8].

Sądzę, że Business Process Reengineering jest koncepcją, która jest komplementarna do współczesnych koncepcji zarządzania, akcentujących potrzebę ciągłego doskonalenia oraz tworzenia kultury organizacji, stanowiącej podstawę wysokiej jakości produkcji oraz nowoczesnych więzi z partnerami.

Reengineering jest inną drogą, która umożliwia wyrównanie, a nawet przewyższenie światowych standardów efektywności, które osiągnięto w wyniku stosowania koncepcji zarządzania akcentujących ciągłe doskonalenie, lecz nie w wyniku wieloletnich starań, a doskonale opracowanego, wpieranego przez najwyższe kierownictwo, jednorazowego programu o charakterze strategicznym.

Uważam, że różnice pomiędzy reengineeringiem (innowacje procesów), a dotychczasowymi koncepcjami (doskonalenie procesów) wynikają właśnie z komplementarności z nowoczesnymi koncepcjami, a nie ze sprzeczności pomiędzy ich celami, czy założeniami. Różnice te przedstawione są w tabeli 4.

Istota Business Process Reengineeringu

Czym więc jest Business Process Reengineering? Według klasycznej definicji Michaela Hammera oraz Jamesa Champy'ego **BPR jest przemyśleniem od podstaw i radykalnym przebudowaniem procesów działalności po to, aby osiągnąć istotną poprawę w krytycznych, podstawowych miarach sprawności, takich jak koszt, jakość, poziom obsługi i szybkość** [7].

Podstawa porównania	Doskonalenie procesów	Innowacje procesów
Zakres zmian	Niewielki	Radykalny
Punkt wyjścia	Istniejące procesy	Projektowanie od nowa
Częstotliwość zmian	Jednorazowe / ciągłe	Jednorazowe
Wymagany czas	Długi	Krótki
Inicjatywa	Oddolna	Odgórna
Typowy zakres	Wąski, wewnątrz funkcji	Szeroki, ponadfunkcjonalny
Ryzyko	Umiarkowane	Wysokie
Typ zmian	Kulturowe	Kulturowe / Strukturalne

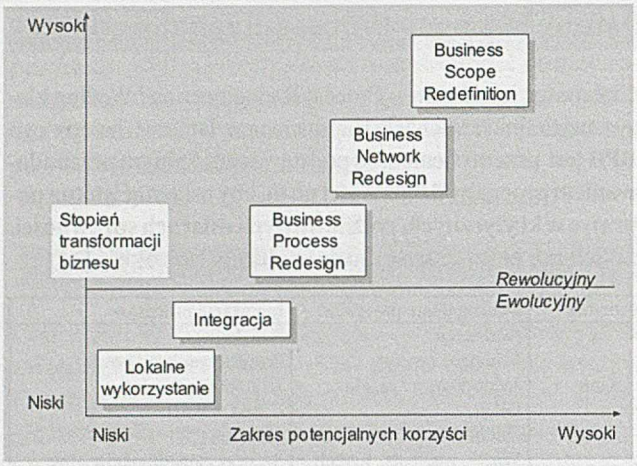
Tabela 4. Ciągłe doskonalenie procesów a innowacje procesów
Źródło: Thomas H. Davenport: Process Innovation: Reengineering Work through Information Technology, Harvard Business School Press, Boston 1993, s. 11.

Autorzy tej definicji wyróżniają cztery kluczowe słowa:

1. podstawowy – BPR podważa wszystkie założenia, na których opiera się biznes, tj. dotychczasową strategię firmy oraz procedury operacyjne. Podważanie ich słuszności umożliwia poszukiwanie przełomu w efektywności działania, poprzez eliminację błędnych strategii oraz zbędnych lub błędnych procedur operacyjnych.
2. radykalny – reengineering, gdy nadarzy się okazja do organizacyjnej innowacji tworzy nowe procesy, a nie usprawnia istniejące,
3. istotny – wzrost efektywności ma na celu znaczny wzrost sprawności, a nie jedynie marginalny jej przyrost, podobny do tego, jaki można uzyskać w wyniku technik ciągłego doskonalenia.
4. procesy – przełamuje się dotychczasowe podziały funkcjonalne i likwiduje w ten sposób nieefektywność będącą skutkiem przekazywania pracy pomiędzy wyspecjalizowanymi jednostkami organizacyjnymi i pracownikami. Praca winna być zorganizowana wokół procesów, a nie wokół indywidualnych zadań.

Tak więc reengineering jest jednym z podejść do doskonalenia działalności, który charakteryzuje radykalny zakres proponowanych zmian i potencjalnie duże korzyści oraz jego jednorazowy charakter. Jednak rozmiar korzyści, które może zyskać przedsiębiorstwo wiąże się również z bardzo wysokim ryzykiem. W celu ograniczenia ryzyka przedsiębiorstwo musi zdobyć wiedzę, dotyczącą metod przeprojektowania procesów oraz ich wdrażania. Dla sukcesu projektu niezbędna jest wiedza z zakresu informatyki oraz znaczne zasoby finansowe i ludzkie.

Jest to technika, która stosowana we właściwy sposób przez przedsiębiorstwa określonego typu, może stać się znakomitym narzędziem stanowiącym o sukcesie rynkowym organizacji. Zasadniczo takie pojmowanie reengineeringu nie jest kwestionowane. Kontrowersje wzbudzają raczej proponowane przez wielu autorów i wiele firm metody projektowania nowych procesów, sposoby wdrożenia ich w firmach oraz praktyczna skuteczność koncepcji.



Rys. 3. Indukowane technologią informacyjną transformacje biznesu.
Źródło: Michael S. Scott Morton - editor: The Corporation of the 1990s: Information Technology and Organizational Transformation, Oxford University Press 1991, New York.

Koncepcja ta po raz pierwszy zwróciła uwagę menedżerów na technologie informatyczne jako potencjalny kluczowy czynnik, umożliwiający dokonanie tak istotnych zmian. Właściwie do tej pory żadna koncepcja zarządzania nie wykorzystywała informatyki w sposób strategiczny. Roli technologii informatycznych w tej koncepcji poświęcony będzie jeden z kolejnych artykułów cyklu.

Miejsce reengineeringu wśród nowoczesnych koncepcji zarządzania

Przeprojektowanie procesów działalności nie jest jedynym programem, który wykorzystuje technologie informatyczne. Niektórzy autorzy twierdzą, że nie jest również najszerszym, gdyż obejmuje jedynie jedno przedsiębiorstwo. W nowoczesnym ujęciu możliwe jest również przeprojektowanie całego łańcucha dostaw. Na rysunku 3 przedstawiono możliwe oparte na technologiach informatycznych, koncepcje transformacji organizacji oraz stopień potencjalnych zmian.

Inną próbą umiejscowienia reengineeringu na tle nowoczesnych koncepcji jest propozycja podziału tych koncepcji ze względu na ich czas trwania i powtarzalność oraz na przewidywany zakres zmian. Zależności te ukazuje tabela 5.

❖ ❖ ❖

Podsumowując przeprowadzoną dyskusję reengineering jest koncepcją komplementarną w stosunku do dotychczasowych filozofii zarządzania. Pierwszą, która koncentruje się na procesach, radykalną w skali potencjalnych zmian, przekształcającą organizację z funkcjonalnej na procesową, kierującą się wynikami procesów, podkreślającą rolę technologii informatycznych.

Porównanie koncepcji Business Process Reengineeringu z innymi nowoczesnymi koncepcjami przedstawia tabela 6.

Kontekst Wynik	Projekt / jednorazowe	Ciągłe udoskonalanie / trwałe
Drobne udoskonalanie	• Analiza wartości czynności • Analiza wartości sfery nieprodukcyjnej • Analiza wartości procesów	• Kompleksowe zarządzanie jakością • Udoskonalanie procesów działalności • Rachunek kosztów działań
Radykalne innowacje	Innowacje procesów (reengineering, business process redesign)	Nic nie znaczące

Tabela 5. Możliwe podejścia do doskonalenia działalności
Źródło: Thomas H. Davenport: Process Innovation: Reengineering Work through Information Technology, Harvard Business School Press, Boston 1993, s. 24

Element	Kompleksowe zarządzanie jakością (TQM)	Just-In-Time	Inżynieria równoległa (SE)	Time Compression Management / Fast Cycle Response	Business Process Redesign / Reengineering
Zorientowanie	Jakość	Zredukowane zapasy Podwyższona sprawność	Redukcja czasu projektowania i wdrożenia do produkcji Wyższa jakość	Redukcja czasu (czas = koszt)	Procesy, minimalizacja czynności nie dodających wartości
Skala zmian	Ciągłe udoskonalanie	Ciągłe udoskonalanie	Radykalne	Radykalne	Radykalne
Organizacja	Wspólne cele w poprzek funkcji	Komórki i praca zespołowa	B+R oraz Produkcja pracują jako jeden zespół	Oparte na procesach	Oparte na procesach
Klienci	Satysfakcja wewnętrznych i zewnętrznych	Inicjator działań „ciągnie" produkcje	Wewnętrzne partnerstwo	Szybka odpowiedź	Sterowanie „wynikami"
Procesy	Uproszczone, poprawione, mierzone w celu kontroli	Przepływy pracy / efektywność przepływów	Jednoczesny rozwój produktu przez B+R i produkcję	Redukcja czasu w procesach	„Idealne" lub o poprawionej p.łynności
Techniki	Mapy procesów, benchmarking, samoocena, statystyczna kontrola procesów, diagramy	Wizualizacja, kanban, małe partie, szybkie przestawienia produkcji	Grupy zadaniowe, CAD / CAM	Mapy procesów, benchmarking	Mapy procesów, benchmarking, samoocena, systemy informacyjne / technologie informatyczne, kreatywność

Tabela 6. Porównanie filozofii zarządzania
Źródło:Joe Peppard, Philp Rowland: The Essence of Business Process Re-engineering, Prentice Hall 1995, New York, p. 16.

Literatura

[1] Michael E. Porter: What Is Strategy?, Harvard Business Review, November-December 1996, p. 61
[2] Greg Bounds, Lyle Yorks, Mel Adams, Gipsie Ranney: Beyond Total Quality Management. Toward the Emerging Paradigm, McGraw-Hill 1994, New York, p. 17
[3] Ricky W. Griffin: Podstawy zarządzania organizacjami, PWN 1996, Warszawa, ss. 661-664
[4] Peter F. Drucker: Innowacje i przedsiębiorczość. Praktyka i zasady, PWE 1992, Warszawa
[5] Michael Hammer: Reengineering Work: Don't Automate, Obliterate, Harvard Business Review, July-August 1990, p.104-112 oraz Thomas H. Davenport, James E. Short: The New Industrial Engineering: Information Technology and Business Process Redesign, Sloan Management Review, Summer 1990, p.11-27
[6] Michael Hammer, James Champy: Reengineering the Corporation. A Manifesto for Business Revolution, HarperCollins Publishers 1993, New York
[7] Michael Hammer, James Champy: Reengineering the Corporation. A Manifesto for Business Revolution, Nicolas BrealeyPublishing 1993, London, p. 49
[8] Thomas H. Davenport, James E. Short: The New Industrial Engineering: Information Technology and Business Process Redesign, Sloan Management Review, Summer 1990, p.11-27 i Thomas H. Davenport: Process Innovation: Reengineering Work through Information Technology, Harvard Business School Press, Boston 1993.

Spójna sieć komputerowa przedsiębiorstwa

Część II

Zdzisław Wrzeszcz

Kanał transmisji komunikatów - cechy ogólne

Kanał jest pewnym zasobem dzielonym przez procesy, które zabiegają o dostęp do niego. Zapewnienie procesowi zadaniowemu *dostępu* do kanału jest więc pierwszym mechanizmem wymagającym zdefiniowania.

Wprowadzany do kanału komunikat powinien być kierowany do określonego procesu – odbiorcy. Wymóg ten jest spełniony przez przypisanie każdemu komunikatowi *adresu odbiorcy* (nie jest to jedyny sposób ustalania związku pomiędzy nadawcą a odbiorcą, okazał się jednak niezwykle skuteczny i jest obecnie szeroko stosowany w tzw. „przełączaniu pakietowym"). Procesy zadaniowe muszą mieć zatem zdolność wprowadzania adresu do pakietu i rozpoznawania adresu tam umieszczonego.

Komunikat powinien mieć również *adres nadawcy*, by umożliwić procesowi – odbiorcy ustalenie sposobu wykorzystania komunikatu.

W przypadku przedsiębiorstw małych i zwartych struktura kanału jest dość prosta. Jednak w przypadku przedsiębiorstw o dużej złożoności technologicznej [9] oraz dużej *rozległości* przestrzennej [4], struktura kanału wydatnie komplikuje się i powstaje kwestia *wyboru trasy* jaką ma przebyć komunikat.

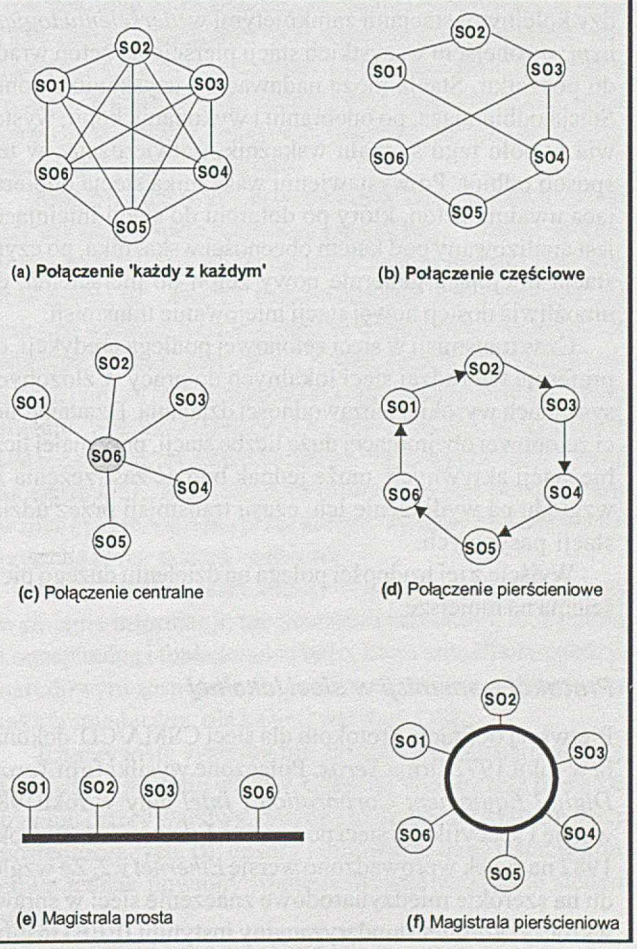
Kanał transmisyjny dla przedsiębiorstwa zwartego

Jako środek transmisji komunikatów spójnej sieci przedsiębiorstwa małego lub sieci wydziałowej przedsiębiorstwa rozległego dostateczny jest kanał występujący w tzw. *sieci lokalnej* [1],[3],[4]. Łącze takiej sieci może mieć fizyczną postać pary przewodowej lub kabla koncentrycznego; nośnikiem komunikatu jest wtedy sygnał elektryczny. Kanałem może też być światłowód [8], nośnikiem komunikatu będzie wtedy impuls świetlny z diody laserowej. O wyborze rodzaju nośnika transmisji decydują wymagania stawiane pojemności i szybkości transmisji komunikatów.

Topologia sieci lokalnej

Postać topologiczna kanału sieci lokalnej może być różna. Połączenie pełne każdej stacji z każdą (Rys. 1a) staje się niedogodne już przy liczbie kilku stacji: splot kabli biegnący we wszystkich kierunkach nie sprzyja niezawodności działania systemu i podnosi koszt instalacji. Połączenie częściowe (Rys. 1b) obniży koszt instalacji lecz podwyższy czas przesyłania niektórych komunikatów. Połączenie centralne (Rys. 1c), ze stacją centralną uczestniczącą w przesyłaniu wszystkich komunikatów, obniża czas przesłań lecz podwyższa koszt instalacji. Połączenie pierścieniowe (Rys. 1d) jest rozwiązaniem kompromisowym. Szybszym jednak rozwiązaniem może okazać się połączenie w postaci szyny lub magistrali, gdyż tu stacje nie stoją na drodze przesyła-

Rys. 1. Topologia połączeń sieci lokalnej



nia komunikatu; możliwe są dwa warianty: szyna liniowa (Rys. 4e) i szyna pierścieniowa (Rys.4f). Rozwiązania takie charakteryzują się też wysoką niezawodnością, gdyż awaryjność kanału jest mniejsza od awaryjności stacji.

Mechanizmy dostępu

W kanale o topologii *magistrali* stacja podejmująca próbę nadania komunikatu musi uprzednio stwierdzić brak innych sygnałów w ośrodku transmisyjnym. Mechanizm ten jest znany pod skróconą nazwą CSMA/CD (Carrier Sense Multiple Access with Collision Detection). Po ewentualnym stwierdzeniu istnienia obcego sygnału stacja czeka pewien czas i ponawia badanie. Po uzyskaniu dostępu i nadaniu własnego sygnału, transakcję komunikacyjną uznaję się za zakończoną – system odbierający nie wysyła potwierdzenia. Dodatnią stroną sieci CSMA/CD jest jej wysoka wydajność dla przypadku dużej liczby systemów w sieci i małej liczby systemów aktywnych: systemy pasywne nie ograniczają działania aktywnych. Jednak przy dużej liczbie systemów aktywnych rośnie liczba kolizji a stąd też czas oczekiwania systemów na transmisję. Czas transmisji informacji w sieci tego typu nie podlega predykcji, co w połączeniu z brakiem potwierdzenia odbioru sprawia, że sieć tego rodzaju jest rozwiązaniem zawodnym. Praktyczna szybkość transmisji sieci CSMA/CD wynosi 10 Mb/s a rozmiar transmitowanego bloku danych 1500 bajtów.

Mechanizm dostępu w sieciach *drugiego rodzaju* jest oparty na prawie dostępu do nadawania, określanym też mianem *żetonu* (token). Żeton taki jest przesyłany pomiędzy kolejnymi stacjami zamkniętymi w *pierścieniu logicznym;* po obejściu wszystkich stacji pierścienia żeton wraca do początku. Stacja może nadawać po uzyskaniu żetonu. Stacja odbierająca, po odebraniu i wykonaniu kopii, wystawia w polu tego sygnału wskaźnik, potwierdzając w ten sposób odbiór. Po wystawieniu wskaźnika stacja odbierająca uwalnia żeton, który po dotarciu do stacji inicjującej jest analizowany pod kątem obecności wskaźnika, po czym stacja inicjująca generuje nowy żeton do pierścienia, co umożliwia dostęp nowej stacji inicjowanie transmisji.

Czas transmisji w sieci żetonowej podlega predykcji, co preferuje ten rodzaj sieci lokalnych do pracy w złożonych systemach wysokiej niezawodności działania. Działanie sieci żetonowej obejmującej dużą liczbę stacji, przy małej liczbie stacji aktywnych, może jednak budzić zastrzeżenia ze względu na wydłużenie ich czasu transmisji przez udział stacji pasywnych.

Wyjście z tej trudności polega na dzieleniu dużego pierścienia na mniejsze.

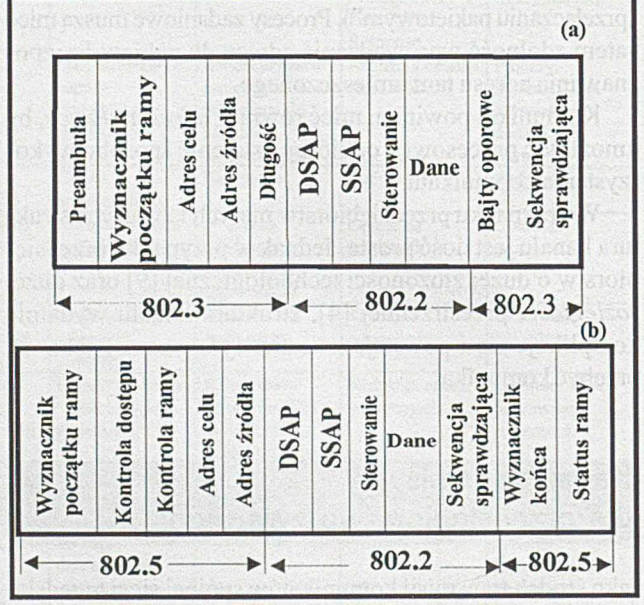
Protokół transmisji w sieci lokalnej

Pierwszej definicji protokołu dla sieci CSMA/CD dokonała w roku 1972 firma *Xerox*. Połączone wysiłki firm *Xerox*, *Digital Equipment Corporation* i *Intel* dały w roku1980 wersję I specyfikacji sieci pod nazwą *Ethernet* v. 1 a w roku 1982 na rynek wprowadzono wersję *Ethernet v.2*. Ze względu na szerokie międzynarodowe znaczenie sieci w sprawę wkroczył komitet standaryzacyjny instytutu IEEE (Institute of Electrical and Electronic Engineers); raporty tego instytutu są dostarczane do ANSI (American National Standards Institute) a stąd do międzynarodowej organizacji standardów ISO (International Standards Organization). Produktem działania IEEE w zakresie sieci CSMA/CD była specyfikacja 802.3 jako część szerszej specyfikacji sieci lokalnych LAN: IEEE 802.

Podobnie było też z siecią żetonową. Choć początki prac można wyśledzić już w roku 1969, to dopiero w roku 1982 *IBM* opracował specyfikację pod nazwą *Token Ring*. Wersja *IBM* była podstawą dla komitetu IEEE do opracowania specyfikacji IEEE 802.5, wydanego w roku 1985. Wydanie specyfikacji 802.5 spotkało się ze szczytem rozwojowym komputerów osobistych PC; połączenie tych rozwiązań stworzyło podstawę dla rozwoju przemysłowych i prywatnych sieci lokalnych.

Informacja przesyłana pomiędzy komputerami współpracującymi w sieci lokalnej o topologii magistrali lub pierścienia przyjmuje formę ramy złożonej z pól, w których zawarta jest informacja o określonej funkcji. Pola takie mogą mieć ustaloną pojemność, są też w określony sposób wzajemnie usytuowane. Ustalenia dotyczące takiej ramy mają charakter protokółu. Rysunki Rys. 2a i Rys.2b opisują zawartość ram IEEE 802.3 i 802.5. A oto treści poszczególnych pól:

Rys. 5. Struktura ramy IEEE 802.3 i 802.5



Protokół współpracy sieci lokalnych

Czy sieć CSMA/CD może współpracować z siecią żetonową? Protokoły ich realizacji firmowych są zdecydowanie różne i nie zakładają takiej współpracy. Na pomoc przychodzi wersja instytutowa; IEEE opracował specyfikację 802.2, która umożliwia taką współpracę. W protokole tym funkcje LAN są ujęte w dwie grupy:

LLC - kontrola połączenia logicznego (*Logical Link Control*),

MAC - kontrola dostępu do ośrodka transmisyjnego (*Media Access Control*).

LLC pełni funkcję sprzęgu, który określa jak sieć danego rodzaju (żetonowa lub CSMA/CD) współdziała z protokołami sieciowymi wyższego poziomu (np. z protokołami SNA lub IPX). W sprzęgu LLC można wyróżnić dwa ważne pola:

SSAP: określające źródłowy punkt dostępu do obsługi (*Source Service Access Point*),

DSAP: określające docelowy punkt dostępu do obsługi (*Destination Service Access Point*).

Pola te zawierają logiczną identyfikację sieci wysyłającej komunikat z żądaniem usługi i sieci mającej świadczyć tę usługę. Jeśli jakiś system odbiera pakiet adresowany do niego, to w pierwszej kolejności czyta pole DSAP i uruchamia program obsługi danego polecenia. Zawartości pola SSAP i DSAP mogą wynosić np.: 00 - zawartość pusta, 04 - operacja wykonywana przez stację SNA, 06 - protokół IP (TCP/IP), itd.

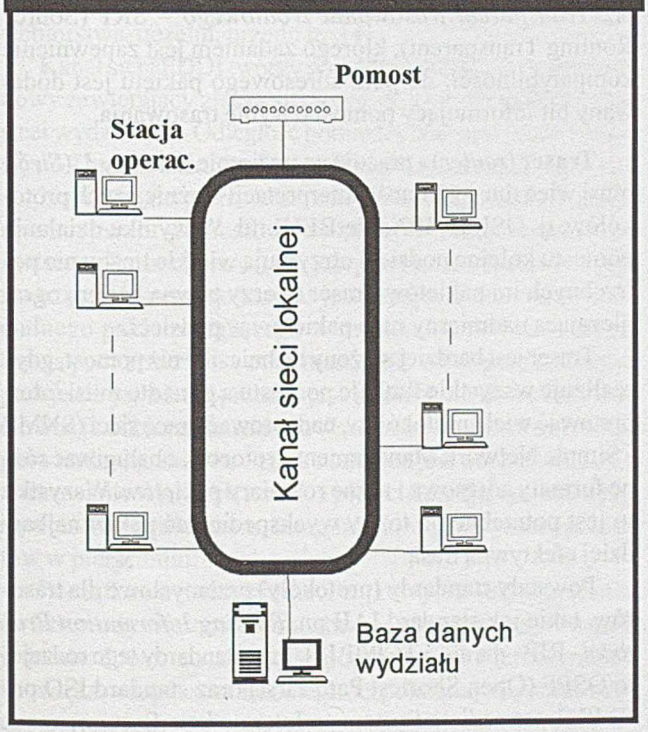
MAC – określa metodykę dostępu, formę i parametry sygnału oraz warunki fizyczne podłączenia do ośrodka transmisyjnego danej sieci.

Protokół 802.2 jest składnikiem obydwu wcześniej opisanych protokółów (patrz Rys.2).

Struktura sieci SSK przedsiębiorstwa zwartego

Opisane tu zasady tworzenia stacji zadaniowych (procesorów) i kanału transmisji komunikatów, w postaci kanału sieci lokalnej dostarczają nam prostego obrazu struktury spójnej sieci przedsiębiorstwa zwartego (struktura SSK_Z – Rys. 3).

Rys.3. Spójna sieć komputerowa przedsiębiorstwa zwartego lub wydziału



Kanał transmisyjny sieci przedsiębiorstwa rozległego

W przypadku gdy rozpiętość obszaru zajmowanego przez przedsiębiorstwo przekracza wartość dopuszczalną dla kanału typu sieć lokalna, stosowanie kanału sieci lokalnej napotyka trudności. Ogólnie biorąc, są dwa rozwiązania: pierwsze, to przyjęcie technologii sieci o lepszych parametrach, tj. pozwalającej na łączenie stacji bardziej odległych; drugie, to bazowanie na sieci z kanałem lokalnym jak dla przedsiębiorstwa zwartego (SSK_Z) przypisując taką sieć do wydziałów przedsiębiorstwa rozległego, a następnie łączenie takich członów SSK_Z w jedną spójną sieć rozległą (SSK_R).

Drugie podejście wydaje się bardziej atrakcyjne, choćby ze względu na możliwość etapowej realizacji sieci; nowe technologie sieci wymagają też nowych rozwiązań wszystkich elementów sprzętowych i programowych, towarzyszących nowej organizacji sieci.

Zestaw protokółów OSI

Rozwój technologii elektronicznych ułatwił produkcję sprzętu i oprogramowania sieciowego wielu producentom, to z kolei owocowało licznymi, własnymi (często zastrzeżonymi) protokółami transmisji; w konsekwencji projektant tworzący sieć przedsiębiorstwa nie może liczyć na jednorodność protokółów środków sieciowych.. By zapanować nad tą sytuacją Międzynarodowa Organizacja Standardów ISO (International Standards Organization) opracowała w roku 1984 zespół standardów pod wspólną nazwą modelu odniesienia OSI (Open System Interconnection Reference Model), dzięki któremu możliwa staje się komunikacja stacji zarówno w obrębie danej sieci lokalnej, jak też i pomiędzy stacjami w różnych sieciach.

Model ten ma strukturę warstwową. Dana warstwa świadczy usługi warstwie położonej tuż nad nią. Warstwa standaryzuje sposoby formułowania informacji na danym poziomie. Postępując od ośrodka transmisyjnego, najniższy poziom zajmuje warstwa fizyczna, nad nią bezpośrednio znajduje się warstwa łączenie danych itd. Każda z warstw definiuje jednostkę danych przeznaczoną do komunikacji pomiędzy warstwami. W gruncie rzeczy są to dane aplikacji, których blok jest poszerzany o informację niezbędną do wypełnienia funkcji na rzecz OSI. Ta dodatkowa informacja jest powiększana w miarę przechodzenia przez kolejne warstwy i będzie wykorzystana przez system - odbiorcę.

Model OSI określa zasady współdziałania stacji przy wymianie informacji; nie dostarcza protokółu, lecz pewną konceptualną i funkcjonalną bazę, która umożliwia międzynarodowym zespołom ekspertów niezależne opracowanie takich standardów dla każdej warstwy OSI.

Rzeczywiste systemy nie zawsze wymagają realizacji wszystkich siedmiu warstw. Czasami wystarczają pierwsze trzy lub cztery warstwy, dotyczy to przede wszystkim stacji pośredniczących, (tj. trasery - omówione dalej); stacje końcowe jednak powinny implementować wszystkie siedem warstw.

Porządek wprowadzany przez strukturę modelu odniesienia OSI przemawia do wyobraźni twórcy sieci, jednak rynek środków sieciowych nie dojrzał jeszcze do pełnej jego akceptacji. Za wyjątkiem standardów częściowych (tj. X.400), standard OSI jako całość traci zainteresowanie realizatorów na rzecz innych, np. TCP/IP [4][10]. Wciąż jednak może stanowić bazę do dyskusji nad komunikacją pakietową. Całość opisuje następująca tabela:

Tabela 1. Warstwy modelu odniesienia OSI

Nazwa warstwy	Przedmiot definicji
7. Aplikacja	Obsługa aktualnych aplikacji, poczta, zarządzanie bazami danych, serwery, polecenia systemu operacyjnego
6. Prezentacja	Formaty danych, zasady emulacji terminali, zasady transferu plików
5. Sesja	Połączenie współpracujących. aplikacji, tworzenie i niszczenie sesji transmisyjnych, synchronizacja w trakcie sesji
4. Transport	Zwielokrotnienie połączenia transportowego dla danego połączenia, ustalanie i naprawa kolejności pakietów
3. Sieć	Sposób przekazywania informacji pomiędzy źródłem, węzłami pośrednimi i węzłem przeznaczenia
2. Łączenie danych	Metody dostępu do ośrodka transm., definicja , metody tworzenia, wysyłania i odbierania pakietu,
1. Fizyczna	Fizyczne podłączenie urządzenia (kabel, skrętka, światłowód, łączówka), parametry sygnału jedynki i zera., kierunek transmisji (dupleks, pół dupleks)

Łączniki sieciowe

Protokoły określają zasady dostępu do kanału i transmisji informacji (komunikatu) w tym kanale; niezbędne są więc pewne środki techniczne posiadające zdolności realizacji tych zasad. Środki te nazywamy łącznikami sieciowymi. Łączniki różnią się, zależnie od repertuaru wykonywanych funkcji.

Powtórnik (ang. repeater). Potrzeba łączenia sieci pojawia się już na poziomie wydziału: trzeba łączyć kolejne segmenty sieci lokalnej tego samego standardu. Najprostszym połączeniem jest połączenie na poziomie warstwy fizycznej. Służące do tego celu złącze międzysieciowe, zwane tu powtórnikiem, jest po prostu wzmacniaczem sygnałów. Wzmocnione sygnały są przekazywane do kolejnego segmentu SSK_Z, będącego siecią odpowiedniego wydziału przedsiębiorstwa. Segmenty te legitymują się tym samym protokółem.

Wzmacniacze mogą być wyposażone w porty (dwa lub więcej). Umożliwia to :

- dołączanie gałęzi Ethernet z różnym przewodnikiem, np. 10Base-5 - tzw. gruby Ethernet, 10Base2 - cienki Ethernet, 10BaseT - nieekranowana skrętka itd.,
- powiększanie odległości w sieci Token Ring, także przy różnych kablach.

Ze względu na poziom łączenia - (warstwa *1.Fizyczna*) - złącze nie ma wpływu na ruch pakietów; we wszystkich częściach połączonej sieci ruch ten jest identyczny.

Pomost (*bridge*). Złącze międzysieciowe tego typu działa na poziomie warstwy *2.* (*Łączenie danych*) a właściwie na dolnej podwarstwie MAC (patrz punkt *Współpraca sieci lokalnych*).

Zadaniem pomostu jest realizacja łączenia sieci lokalnych wydziałowych lub lokalnej sieci przedsiębiorstwa z sieciami oddalonych oddziałów; łączone sieci mogą być różnych standardów (TokenRing, Ethernet, ..), pomost może być wyposażony w łączówki dla różnego okablowania fizycznego tj. kabel koncentryczny gruby, cienki, skrętka itd.

Znane są dwie różne metody przesyłania pakietów za pomocą pomostów, metody te wprowadzili producenci sprzętu na rynek amerykański. Pierwsza z metod opracowana przez IBM stosuje protokół z *trasowaniem źródłowym.* Źródło wysyła pakiet rozpoznawczy. Po przebyciu drogi pakiet rozpoznawczy pozyskuje zapis trasy dla danej sesji. Następuje ustalenie sesji, każdy pakiet sesji pozyskuje zapis trasy, zapis ten prezentuje napotkanym pomostom.

Trasowanie źródłowe jest stosowane głównie w sieciach pierścieniowych, natomiast w sieciach typu Ethernet stosuje się przezroczyste przekazywanie przez pomosty. Za podstawę trasowania służy wiedza pomostu. Pomost przechowuje tablice z adresami stacji, sprawdza więc czy adres przeznaczenia pakietu odpowiada stacji w sieci przed pomostem, pakiet pozostaje lub jest przekazywany na drugą stronę pomostu. Pomosty biorą pod uwagę nie tylko adresy ale też miejsce przeznaczenia, przeciążenie tras, uszkodzone pomosty itd.; wszystkie te informacje pomost wykorzystuje do wyznaczenia optymalnej trasy. Oznacza to jednak, że kolejne pakiety sesji mogą być wysyłane *różnymi trasami* zmieniając kolejność przybycia do punktu przeznaczenia; wiąże się to zazwyczaj z błędem.

Komitet IEEE 802 opracował standard dla pomostów pn. *Transparent trasowania źródłowego* – SRT (Source Routing Transparent), którego zadaniem jest zapewnienie kompatybilności: do pola adresowego pakietu jest dodawany bit informujący pomost o trybie trasowania.

Traser (*router*) - pracuje na poziomie warstwy *3.* (*Sieć*), musi więc mieć zdolność interpretacji ważniejszych protokółów, tj. OSI, IP, IPX, NetBEUI itd. W wyniku działania pomostu kolejne podsieci otrzymują wielkie liczby nie potrzebnych im pakietów; traser tworzy pewną barierę ograniczającą nadmierny ruch pakietów w podsieci.

Traser jest bardziej złożony technicznie niż pomost, gdyż realizuje wszystkie funkcje pomostu a ponadto musi interpretować wiele protokółów, nadzorować pracę sieci (SNMP - Simple Network Manegement Protocol), obsługiwać różne formaty adresowe i różne rozmiary pakietów. Wszystko to jest potrzebne po to, by wyekspediować pakiet najbardziej efektywną trasą.

Powstały standardy (protokóły) przemysłowe dla traserów, takie jak standard IAB pn. *Routing Information Protocol* - RIP - dla sieci TCP/IP [7]. Inne standardy tego rodzaju, to OSPF (Open Shortest Path First) oraz standard ISO pn. IS-IS (Intermediate-System-to-Intermediate-System).

Koncentrator (*hub*) - jest to złącze sieciowe pracujące pierwotnie na poziomie warstwy 1. i 2. Jednak rozwój technologii wytwarzania urządzeń cyfrowych umożliwił producentom wzbogacenie urządzenia o funkcje występujące w pomostach, traserach i *monitorach sieci* a także o inne funkcje niezbędne dla wyższych warstw protokółu.

Wspomniany rozwój technologii odbija się też korzystnie na unifikacji rozwiązań złącz sieciowych: obserwuje się tendencję do utworzenia *złącza sieciowego jednego, uniwersalnego rodzaju*, integrującego wszystkie niezbędne funkcje, tj. trasowanie wielokanałowe, ustalanie topografii złącz zgodnie z życzeniem twórcy sieci, monitoring sieci, funkcje górnych warstw i być może inne.

Sieć pośrednicząca

Dysponując złączami międzysieciowymi, takimi jak powtórnik, pomost, traser jesteśmy w stanie dokonać dowolnego połączenia pomiędzy sieciami wydziałowymi. Innym rozwiązaniem może być sieć pośrednicząca, która połączy sieci wydziałowe. Łączone w ten sposób podsieci muszą realizować protokół już tylko jednej sieci.

Możliwe są tu dwa rozwiązania: *publiczna* sieć pośrednicząca lub też *prywatna* sieć pośrednicząca. Rozwiązanie pierwsze jest rozwiązaniem tańszym, rozwiązanie drugie zaś pozwala dopasować ten środek pośredniej transmisji do własnych wymagań, zapewnić większe bezpieczeństwo danych transmitowanych i gromadzonych. Przykładem publicznej sieci pośredniczącej może być sieć typu *metropolitalnego* (MAN - metropololitan area network) [7], przegląd realizacji takich sieci można znaleźć np. w pracy [5].

Sieć pośrednicząca FDDI. Specyfikacja FDDI odnosi się do sieci światłodowej (FDDI - fiberoptic distributed data interface), jest standardem amerykańskim pn. ANSI X3T9.5. Przeznaczeniem takiej sieci jest właśnie funkcja sieci pośredniczącej dla sieci typu LAN rozłożonych na obszarze przedsiębiorstwa, uczelni, urzędu itp.[4].

Kanał sieci FDDI tworzy podwójny pierścień światłowodowy zawierający złącza sieciowe, do których są dołączone sieci wydziałowe. Odległość pomiędzy stacjami może wynosić do 2 km, pełna długość wokół pierścienia sięga 100km.

Należące do pierścienia stacje realizują funkcje stacji operacyjnych, pomostów, traserów lub koncentratorów. Podstawą łączenia transmisji jest przełączanie pakietów [6]. Podstawą dostępu do kanału jest przejęcie żetonu, tj. specjalnego pakietu krążącego w pierścieniu podstawowym. Stacja powtarza ramkę otrzymaną, jeśli adres przeznaczenia pasuje do jej adresu; ramka jest kopiowana do buforu stacji. Stacja odbierająca ustawia wskaźnik w polu statusu ramki potwierdzając odbiór. Stacja nadająca jest odpowiedzialna za usuwanie własnych ramek z pierścienia. Stacje, które nie są właścicielami żetonu, biorą udział w przesyłaniu pakietów w pierścieniu.

Początkowo realizacje sieci wykorzystywały wyłącznie światłowody wielomodowe, obecnie są stosowane również jednomodowe, co wybitnie poprawia jakość transmisji i powiększa dopuszczalne odległości pomiędzy złączami międzysieciowymi.

Z racji podobieństwa technologii sieci FDDI powinny być zdolne współpracować z siecią szerokiego obszaru (WAN) SONET [4] i tak jest faktycznie: w USA powstała grupa badawcza zajmująca się odwzorowaniem warstwy fizycznej FDDI na STS-3, tj. sygnał transportu synchronicznego- poziom 1 sieci SONET. Dzięki zaletom rozległej sieci SONET można będzie wydatnie zwiększyć granice komunikacji przedsiębiorstwa, pokonując takie przeszkody terenu jak rzeki, góry itd.

Prace nad FDDI wyszły ze wstępnej fazy, na rynku (USA) są dostępne zestawy układów scalonych do realizacji stacji FDDI dla standardu FDDI. Posuwają się pilnie prace nad wersją FDDI II, która umożliwia również transmisję izochroniczną (sygnały mowy i obrazu telewizyjnego). Wprawdzie transmisja izochroniczna nie jest warunkiem niezbędnym do realizacji procesów wytwórczych, ale może być bardzo pomocna dla procesów kontroli i zarządzania w przedsiębiorstwie.

Poniższa Tabela 2 podaje zestaw danych charakterystycznych dla omawianego standardu, Rys. **4** przedstawia warstwy protokółu FDDI, zaś Rys. **5** zamieszcza rozmiary bajtowe pól ramki.

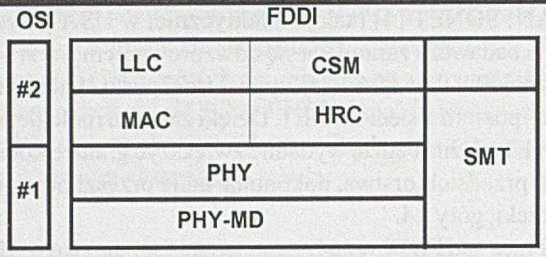
Tabela 2.

Przedmiot	**Opis**
Rodzaj ośrodka transmisyjnego	Podwójny pierścień z dwukierunkową transmisją sygnału
Typ światłowodu	. wielomodowy (62nm) do 2 km pomiędzy złączami . jednomodowy (200nm) do 50km pomiędzy złączami
Pasmo transmisyjne	100 Mbps
Pojemność ramki	4 500 bajtów
Liczba stacji	do 500
średnica obszaru zajętego	100km
Kodowanie sygnału	4B/5B NRZI (1 - przejście, 0 - brak przejścia)
Standardy	
Standard ogólny	ANSI X3T9.5
MAC - kontrola dostępu do ośrodka	X3.139-1987
PHY - warstwa fizyczna	X3.148-1988
PMD - " " zależna od ośrodka	X3.166-1990
SMT - zarządzanie stacjami	w trakcie opracowania

Protokóły TCP/IP. Jest to zestaw protokołów, którego każdy składnik może być użyty niezależnie [7][10]. Zestaw został zaprojektowany do łączenia podsieci w większe formy sieciowe - *intersieci*; jest to więc zestaw protokołów *sieci pośredniczącej*, zgodnie z potrzebami tworzenia sieci przedsiębiorstwa.

Ponieważ podsieci mogą stosować różne technologie, to TCP/IP nie nakłada żadnych ograniczeń technologicznych na niższe, fizyczne warstwy: przenośnikiem sygnału może więc być kanał 802.3, 802.5, X.25 lub inny. Obraz zestawu TCP/IP różni się, zależnie od wykonawcy środków sieciowych.

Rys. 4. Protokóły sieci FDDI



CSM (circuit switching multiplexer) - multiplekser do łączenia obwodów
SMT (station management) - połączenie fizyczne i logiczne stacji, korekcja błędów; w tej warstwie jest też zawarty MIM (media interface management) określający dodtępność obiektów warstwy fizycznej, kontrolę obejść, dostępność ośr. fiz., PCM(physical connention management) - zarządzanie połączeniem fizycznym, SCM (station configuration management) - zarządzanie konfiguracjąstacji.
MAC (media access control) - protokół żetonu uprawniającegonadawanie, format żetonu, format ramki.
HRC (hybrid ring control) -typy transmisji (synchr./asynchr.),komutacja drogi ('tylko raz'/isochroniczna - dla danych isochronicznych).
PHY (physical sublayer protocol) - schemat kodowania danych, zależności czasowe, typy protokołów fizycznych.
PHY-MD (physical sublayer medium dependent) - warstwa fizyczna niższa, określa topologię sieci, typy stacji, właściwości ośrodka trasmis. (modowość, ekran, ..), typ złącz, transmitowane sygnały, parametry nadajników/odbiorników, obejścia.

IP - rezydujący w warstwie *3.Sieć* protokół zapewnia bezpołączeniowe przekazywanie pakietów. Podsieci są połączone ze sobą za pomocą *traserów*; są to sprzętowo - programowe węzły pośrednie, które wyznaczają trasę pakietów (w formacie IP) pomiędzy podsieciami. Pakiet może osiągnąć cel różnymi drogami i nie ma wyznaczonego połączenia przekaźnikowego.

Rys. 5. Format ramki FDDI

Oznaczenia pola 'Status ramki':
E - wskaźnik detekcji błędu
A - " rozpoznania adresu
C - " kopiowania ramki

IPX (Interchange Packet Exchange) - protokół systemu *NetWare* (Novell) służy do wymiany informacji między podsieciami, przekazuje żądania aplikacji, dotyczące usług sieciowych, do sterowników sieciowych,

TCP - protokół w warstwie *4.Transport* organizujący łączoną transmisję pakietów, w pełnym dupleksie 'punkt-punkt'. Jest to połączenie w pełni niezawodne, z wykrywaniem i korektą błędów.

SPX (Sequenced Packet Exchange) - protokół systemu *NetWare* (Novell) do komunikacji dwóch stacji i transmisji pakietów, z gwarancją dostarczenia pakietów i zachowania kolejności w strumieniu.

W warstwach od *5.* do *7.* są umieszczane *usługi aplikacyjne*, takie jak: poczta elektroniczna; podstawowy środek transmisji plików pomiędzy aplikacjami na różnych platformach, dla kilku prostych struktur plikowych (FTP - File Transfer Protocol) w formacie rekordowym, plikowym, ASCII, EBCDIC i binarnym; usługa NFS (Network File Service), która sprawia, że pliki i katalogi na oddalonych platformach zachowują się tak, jakby należały do platformy lokalnej, umożliwia obsługę plików (czytanie, pisanie); SMTP (Simple Mail Transfer Protocol) - przesyłanie poczty pomiędzy dwoma punktami lub przesyłanie do wielu odbiorców w trybie 'gromadź-i-przekaż' i wiele innych usług.

❖ ❖ ❖

Rozpatrzono działania przedsiębiorstwa a następnie dopasowano w sposób spójny środki komputerowe sterujące tym działaniem. Połączone spójnie środki pozwalają skupić uwagę na strukturze sterującej. Logicznym obrazem działania przedsiębiorstwa wraz ze środkami sterującymi jest pewien zbiór spójnych, współbieżnych procesów skoordynowanych. Współbieżne procesy są realizowane przez procesory zadaniowe, które są również odpowiedzialne za wytwarzanie i odbiór sygnałów reprezentujących komunikaty.

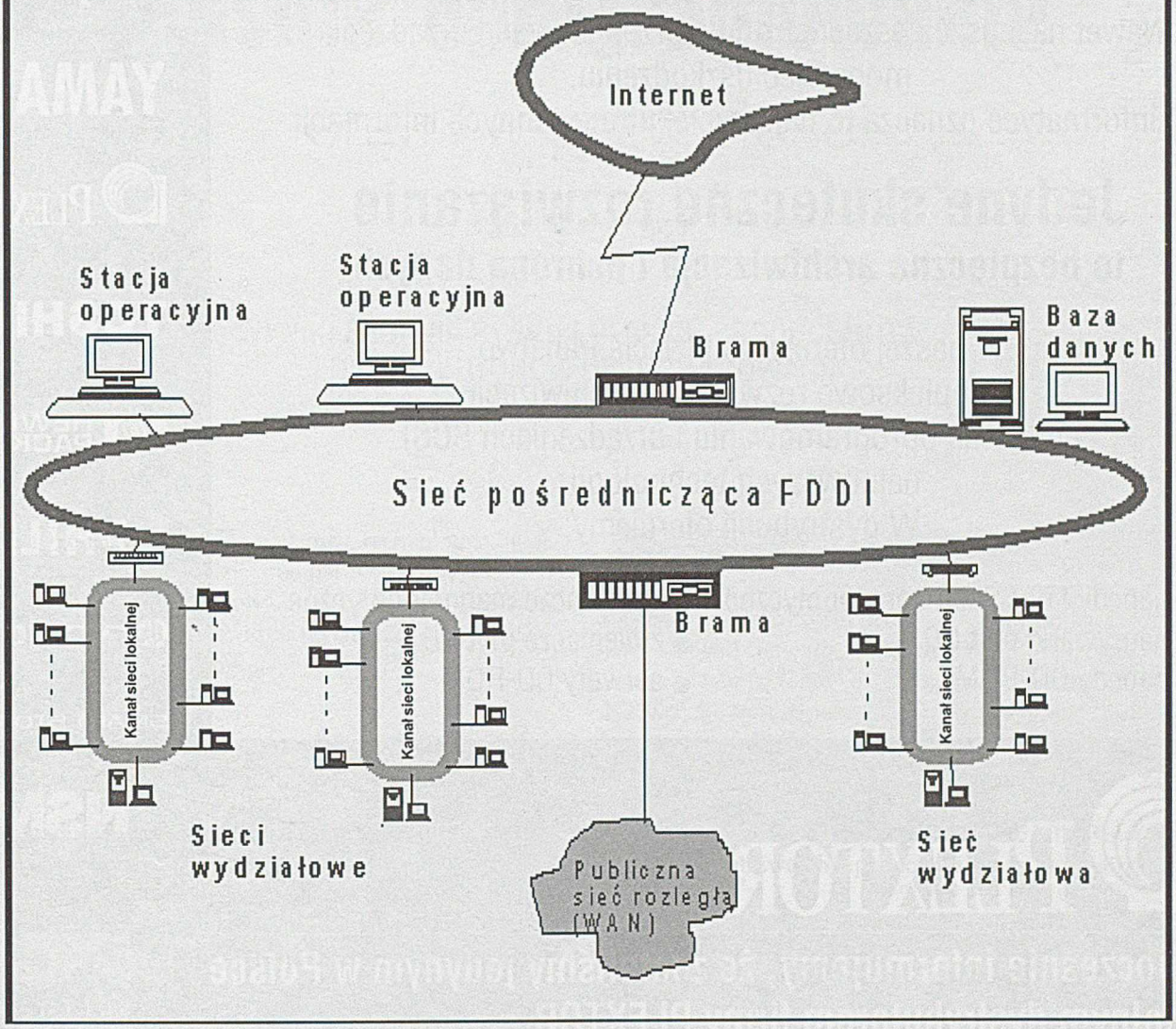
Środkiem transmisji sygnałów pomiędzy procesorami jest kanał transmisji. Stąd ogólnym obrazem sieci komputerowej, spójnej z działaniem przedsiębiorstwa, może być pewna liczba stacji zadaniowych, które wraz z grupami narzędzi zadaniowych są rozmieszczone w pomieszczeniach zadań; stacje te współdziałają za pomocą komunikatów kierowanych poprzez kanał transmisji komunikatów. Środkiem, który może posłużyć do transmisji komunikatów w przedsiębiorstwie zwartym jest kanał sieci lokalnej. Całość tworzy spójną sieć komputerową przedsiębiorstwa zwartego (SSK_Z).

Spójną sieć przedsiębiorstwa rozległego można utworzyć przez połączenie sieci wydziałowych, będących sieciami typu SSK_Z. Środkami pozwalającymi na takie łączenie mogą być standardowe łączniki sieciowe, dostępne na rynku.

W miarę potrzeby (w przypadku niejednorodności technologicznej stacji i kanałów SSK_Z) łącznikiem sieci wydziałowych może być też sieć pośrednicząca wybranej technologii. Można też przyjąć, że sieć pośrednicząca zawiera również wyróżnioną stację interpretującą protokóły TCP/IP. Pozwoli to naszej sieci **SSK_R** komunikować się z siecią **Internet** i korzystać z usług świadczonych przez zainstalowane tam serwery. Sieć **SSK_R** przedsiębiorstwa zyskuje dzięki temu łączność ze światem zewnętrznym, może docierać do zbiorów bankowych, pocztowych, giełdowych itd. Tą drogą kierownictwo przedsiębiorstwa może porozumiewać się z kierownictwem innych przedsiębiorstw; Rys. **9** pokazuje nam strukturę takiej sieci.

Przedstawiono tu nową koncepcję tworzenia sieci komputerowej, sterującej działaniem przedsiębiorstwa; zalety tej propozycji są zamknięte określeniem **spójna** sieć. Z przedstawionych tu zasad realizacji spójnej sieci komputerowej wynika, że tylko część tego nowego rozwiązania wymaga

Rys. 6. Spójna sieć komputerowa przedsiębiorstwa rozległego



opracowania nowych środków wykonawczych; są to sterowniki SOT. Pozostałość natomiast, to kanały, które mogą korzystać z już istniejących rozwiązań sieciowych. Ważnym zadaniem, jak zresztą w przypadku każdej sieci, jest programowanie działania przedsiębiorstwa. W przypadku sieci SSK metodyka programowania jest wysoce jednorodna.

Literatura:

[1] CORRIGAN P.H., A. GUY, ***Building Local Area Networks***, M & T Books, 1992

[2] KLESSIG R.W., ***Overview of Metropolitan Area Networks***, IEEE Comm. Magazine, **24**, 1, 9-15 (1986)

[3] MCCLAIN G.R. (Ed.), ***Handbook of Networking and Connectivity***, AP Professional, Boston, 1994

[4] MINOLI D, ***Enterprise Networking***, Artech House, London, 1993.

[5] MOLLENAUER J.F., ***Standards for Metropolitan Area Network***, IEEE Comm. Magazine, **26**, 1, 15-19 (1988)

[6] SIRBU M.A., L.E. ZWIMPFER, ***Standards Setting for Computer Communication; The Case of X.25***, IEEE Network Magazine, **23**, 3, March 1985

[7] WILDER F., ***A Guide to the TCP/IP Protocol Suite***, Artech House, Boston, London, 1993

[8] WRZESZCZ Z., ***Światłowód jako element sieci transmisji informacji***, NetWorld, 1/1995, IDG Poland SA, Warszawa

[9] WRZESZCZ Z., ***Converter Steelmaking Presented as a Complex Job Computer Control System***. W książce: S. K. GHOSH, A. NIKU-LARI, ***CAD/CAM & FM in Metal Working***. Pergamon Press, Oxford, 1988

[10] XEROX ***System Integration Standard Internet Transport Protocols***. Xerox Corporation, Dec. 1981, Doc. XSIS 028 112.

dr Zdzisław Wrzeszcz jest niezależnym konsultantem.

Bezpieczna archiwizacja danych w technologii CDR i MO

Nie istnieją niezawodne systemy informatyczne.

Nawet najlepsze, wszechstronnie przetestowane urządzenia
mogą ulec uszkodzeniu.
W informatyce oznacza to najczęściej utratę cennych informacji.

Jedyne skuteczne rozwiązanie
to bezpieczna archiwizacja i ochrona danych

W naszej ofercie znajdziecie Państwo
kompleksowe rozwiązania archiwizacji
oparte na oprogramowaniu i urządzeniach SCSI
najnowszych technologii.
W dystrybucji oferujemy:

- napędy i nośniki magnetooptyczne
- nagrywarki płyt CD
- napędy CD-ROM
- zmieniacze magnetooptyczne
- zmieniacze płyt CD
- serwery CD-ROM

FUJITSU
YAMAHA
PLEXTOR
TOSHIBA
HEWLETT PACKARD
PHILIPS
PIONEER
SMART STORAGE
NSM JUKEBOX
iXOS SOFTWARE We Serve our Clients

PLEXTOR

Jednocześnie informujemy, że zostaliśmy jedynym w Polsce oficjalnym dystrybutorem firmy PLEXTOR, znanej z wysokiej jakości CD-ROMÓW SCSI.

Serdecznie zapraszamy
do odwiedzenia naszego stoiska
na targach INFOSYSTEM '97
Hala 23
Poziom C
Stoisko 356

KONKURS z brunatnym MIS-iem

W czasie targów:
SUPERKONKURS
SUPERZABAWA

mis
pl. Powstańców Śl. 16/18
53-314 Wrocław
tel./fax (0-71) 732201
732207

Polskie oprogramowanie

teraz i w przyszłości

Janusz Gieruszyński

WYNIKI BADANIA RYNKU POLSKICH PRODUCENTÓW I DYSTRYBUTORÓW OPROGRAMOWANIA

W 1996 roku wykonano badania „Technologia 2000" w celu określenia najbardziej popularnych wśród polskich producentów i dystrybutorów oprogramowania, technologii informatycznych i przewidywanych przez nich trendów w tym zakresie do 2000 roku.

Metodologia badań

Przyjęto formułę otwartego badania ankietowego wśród producentów i dystrybutorów oprogramowania wspomagającego zarządzanie firmą. Wyboru tego dokonano biorąc pod uwagę obszar zainteresowań Stowarzyszenia PRO i firmy Softtutor oraz silną pozycję polskich firm w tej dziedzinie.

Kwestionariusz wysłany do 384 polskich producentów i dystrybutorów oprogramowania zawierał 10 pytań dotyczących środowiska, interfejsu, bazy danych i narzędzi projektowo - programistycznych dla aplikacji tworzonych (lub sprzedawanych) obecnie i przewidywań w tym zakresie na rok 2000. Wypełniający ankietę musieli dokonać wyboru właściwych odpowiedzi z listy wariantów lub wpisać potrzebne informacje.

Informacje podawane w ankietach nie były weryfikowane. Nie ingerowano w treść odpowiedzi na pytania otwarte. Wyniki badania są więc uśrednioną, subiektywną opinią przedstawicieli firm, którzy wypełniali kwestionariusz.

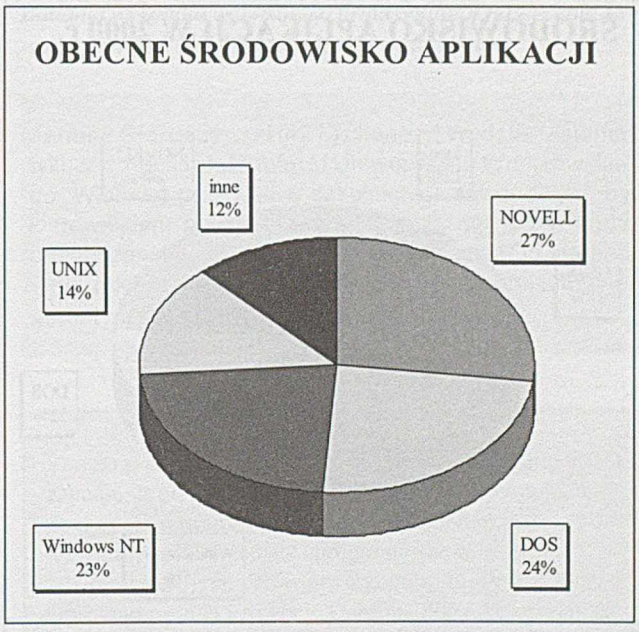
Wśród 53 firm, ktore wzięły udział w badaniu, dominowali producenci i dystrybutorzy zintegrowanych systemów zarządzania, programów finansowo - księgowych i programów kadrowo - płacowych. Nieliczne firmy zajmowały się innymi aplikacjami takimi, jak oprogramowanie biurowe, programy graficzne, prezentacyjne, edukacyjne.

Zdecydowana większość wymieniła jako sprzęt dla swoich obecnych aplikacji komputery klasy PC (przede wszystkim z procesorami typu 486 i Pentium) a ok. 15% firm wskazało inny sprzęt, m.in. DEC Alpha, IBM RISC, SUN.

Technologie 1996 roku

Obecnie ponad 2/3 firm dedykuje swoje aplikacje środowisku Novell Netware. 58,5% firm tworzy aplikacje DOS, a niewiele mniej (56,6%) - oprogramowanie dla Windows NT. Daleko w tyle pozostają firmy dedykujące aplikacje systemowi Unix (34%), a poniżej progu 10% znajdują się producenci i dystrybutorzy programów dla systemów VMS, OS/400 i innych.

Ponieważ jedna firma mogła wskazać kilka środowisk dla swojej (lub swoich) aplikacji, to dokonano analizy liczby środowisk wymienionych w ankietach. Okazało się, że najczęściej wymieniano aplikacje Novell (27%), DOS (24%) i Windows NT (23%). Co siódma aplikacja dedykowana jest systemowi Unix.



Na pytanie o stosowany obecnie interfejs użytkownika w swoich aplikacjach ponad 70% firm wskazało znakowy

i niewiele mniej - graficzny. 30 % deklaruje używanie tylko interfejsu znakowego, 28% - tylko graficznego, a reszta - obu typów interfejsu.

W 1996 roku ponad połowa firm deklarowała, że w ich aplikacjach znajduje zastosowanie baza danych formatu dBase (.dbf). Prawie dwa razy mniej firm wskazywało bazę Access, a niecałe 23% - Btrieve. Mniej więcej co piąta firma wymieniała bazy: Oracle, Sybase lub Informix. Wśród innych baz danych najczęściej wpisywano Microsoft SQL.

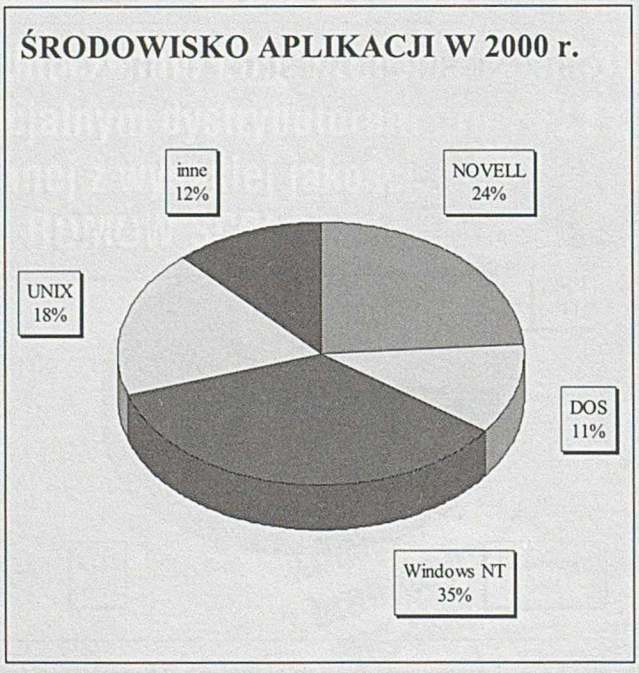
Także analiza ilości wymienionych w ankietach baz danych potwierdziła ranking: 26% ogółu to bazy formatu dbf, 14% Access, po 10% bazy Btrieve, Oracle, Sybase i Informix.

Technologia 1996 roku: DOS / Novell Netware, interfejs znakowy, baza danych formatu .dbf, różnorodne narzędzia projektowo - programistyczne

Odpowiadając na pytanie „Jakie narzędzia projektowo - programistyczne stosujecie przy produkcji aplikacji?" najczęściej wymieniano różne odmiany języka C i Clippera oraz narzędzia Oracle (prawie 35% odpowiedzi). Jest jednak duży rozrzut odpowiedzi, bo udało się sklasyfikować 13 grup narzędzi, a prawie 30% stanowi kategoria „Inne narzędzia", zawierająca bardzo dużo języków, generatorów, bibliotek itp.

Technologie 2000 roku

W przyszłości ponad 90% firm pragnie tworzyć aplikacje dla Windows NT, 64% będzie dedykować oprogramowanie sieci Novell Netware, a prawie 50% systemowi Unix.



Prawie połowa firm, które obecnie wskazują DOS jako środowisko swojego produktu, zapowiada wycofanie się z niego (obecnie 58,5%, w 2000 roku - 30,2%). Podobny wynik otrzymano przy analizie liczby wymienianych przez firmy środowisk aplikacji w 2000 roku: Windows NT 35%, Novell 24%, Unix 18%, a DOS - jedynie 11%.

Jak należało oczekiwać, prawie wszystkie firmy (98,1%) deklarują w najbliższym czasie przejście na graficzny interfejs użytkownika. Jednocześnie ponad 43% firm zapowiada pozostawienie w aplikacjach trybu znakowego. Tylko jedna firma z ankietowanych pozostanie wierna interfejsowi znakowemu - nie stosuje i nie zamierza stosować graficznego!

W 2000 roku ponad połowa firm zapowiada stosowanie bazy Oracle (52,8%), a 41,5% ogółu firm pragnie zastosować w swoich aplikacjach bazę Informix. Wyraźnie spadnie zastosowanie baz formatu dbf, mniej będzie także firm stosujących bazę Access. Nadal wielu zwolenników będzie miała baza Microsoft SQL. Uwagę zwraca także duża liczba firm niezdecydowanych (15%), w których brak ustalenia bazy danych w przyszłych aplikacjach (odpowiedzi typu: trudno powiedzieć, pochodne dbf, dobre i szybkie, nie wiemy itp.)

Język obiektowy będzie stosować 75,5% firm, język 4GL - 43,4%. Standardowe narzędzia projektowo - programistyczne będzie używać w 2000 roku 37,7% firm, co czwarta zapowiada stosowanie własnych narzędzi.

Deklarowany jest niewielki wzrost stosowania C i Oracle, ale także co siódma odpowiedź wskazuje na brak planów na przyszłość (prawie 15% firm nie wie, jakie narzędzia zastosuje w 2000 roku).

Kierunki zmian

W najbliższym czasie oczekiwać należy wyraźnego wzrostu liczby aplikacji dla Windows NT kosztem oprogramowania dla DOS. Pozostałe środowiska utrzymają w zasadzie swój obecny „stan posiadania". Taką tendencję potwierdza też porównanie środowisk aplikacji tworzonych obecnie i zapowiadanych na przyszłość przez firmy, które deklarują odejście od środowiska DOS (28% ogółu firm). Wszystkie one planują w 2000 roku tworzenie aplikacji dla Windows NT.

Nie ma firmy, która zrezygnowałaby z tworzenia oprogramowania dla Windows NT. Aż 36% producentów i dystrybutorów planuje wejść na rynek aplikacji NT w okresie najbliższych trzech lat. Drugim środowiskiem, z którego nikt nie będzie rezygnował, a nawet są nowi chętni do wykorzystania w przyszłości, jest OS 400 - oczywiście w nieporównywalnie mniejszej skali (około 5% firm).

Wyraźny spadek liczby firm dedykujących swoje aplikacje dla środowiska DOS (o połowę) nie wiąże się ze zmianą ilości firm tworzących oprogramowanie dla sieci Novell Netware. Obecnie prawie 68% firm produkuje lub sprzedaje programy pracujące w środowisku Novell NetWare, za trzy lata będzie to czynić 64%. Należy więc założyć utrzymanie

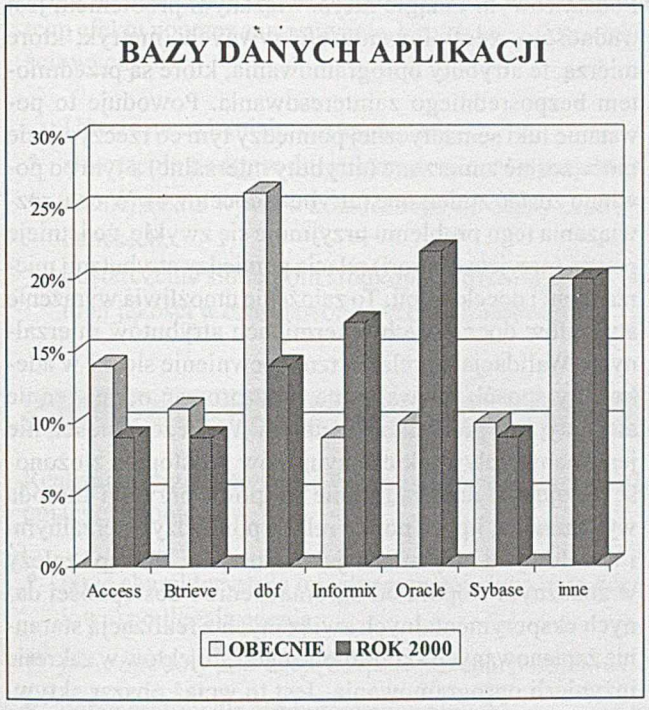
w najbliższych latach dużej liczby producentów oprogramowania dedykowanego sieciom Novell'a. Taką tendencję potwierdza także porównanie liczby wskazanych przez firmy aplikacji: obecnie 27% ogółu stanowią programy dla sieci Novell, w 2000 roku będzie to odpowiednio 24%.

W ciągu najbliższych trzech lat stosunek ilości aplikacji ze znakowym interfejsem użytkownika do ilości aplikacji z interfejsem graficznym powinien zmienić się z wartości 1 : 1 na 1 : 2 . Identycznie przekształcą się proporcje firm stosujących oba typy interfejsów.

Do 2000 roku obserwować będziemy powolne odchodzenie firm od baz danych typu dbf (zmniejszenie mniej więcej o 1/3) przy równoczesnym zdecydowanie większym zastosowaniu baz Oracle i Informix z zachowaniem stałego poziomu stosujących inne bazy. Tendencja jest jednak bardziej złożona. Otóż firmy, które dopiero zapowiadają zastosowanie bazy Oracle (1/3 ogółu), zakładają także utrzymanie stosowanych obecnie typów baz danych (lub ich niewielkie korekty) oraz wprowadzenie baz Oracle i Informix jako nowych. Podobny wynik otrzymamy porównując analogicznie oprogramowanie dla bazy Informix (25% firm).

Tak więc wzrost zastosowania baz danych Oracle i Informix nie jest prostym odpowiednikiem spadku zainteresowania bazami danych typu dbf. Tylko co szósta firma planująca użycie Oracle lub Informix uczyni to „kosztem" baz formatu dBase. Pozostałe planują tworzyć aplikacje z bazą danych Oracle (Informix) jako nowe lub z zachowaniem aplikacji (i baz danych) już istniejących.

Polska firma informatyczna 2000-ego roku będzie tworzyć aplikacje przede wszystkim dla środowiska Windows NT z interfejsem graficznym i obsługą baz danych kilku najważniejszych producentów w oparciu o różnorodne narzędzia projektowo - programistyczne.



Również nie wszystkie firmy odchodzące od baz formatu dBase przechodzą wyłącznie na bazy Oracle lub Informix.

W zakresie narzędzi projektowo - programistycznych wyniki badania nie sygnalizują żadnych zmian w latach 1996 - 2000. Obecnie i w przyszłości bardzo podobne liczby firm wskazują identyczne narzędzia, np. język obiektowy stosuje - ich zdaniem - 71,7% firm, a będzie stosować - 75,5%. Jako „własne" określa obecnie narzędzia projektowo - programistyczne 32,1% firm, a zapowiada stosowanie ich w 1999 roku - 24,5%. Różnice są niewielkie, rzędu kilku procent.

Porównanie liczby i typu wymienionych przez firmy narzędzi także nie ujawniło poważnych zmian w zastosowaniu przez firmy różnych narzędzi projektowo - programistycznych przy produkcji swoich aplikacji. O ile w najbliższym czasie zapowiadany jest niewielki wzrost stosowania C i Oracle, to ilość zastosowań Clippera zmaleje prawie o połowę. Rośnie liczba aplikacji budowanych przy pomocy Power Buildera, całkowicie zanikną aplikacje tworzone w Turbo Pascalu. Zmiany te jednak nie są duże (maksymalna wartość to „spadek" Clippera o 4,5 punktu procentowego) i raczej nie mogą być podstawą do określenia kierunku zmian stosowanych typów narzędzi. Jest to istotne z uwagi na dużą liczbę wariantów odpowiedzi, w tym szczególnie z kategorii „trudno powiedzieć" (np.: jakie narzędzia będą stosowane w 2000 roku).

Badania wykonała warszawska firma Softtutor na zlecenie Stowarzyszenia Polski Rynek Oprogramowania.

Janusz Gieruszyński (42 l.) ukończył Wydział Matematyki, Mechaniki i Informatyki Uniwersytetu Warszawskiego. Wiele lat pracował w szkolnictwie zawodowym, był kierownikiem pracowni przetwarzania danych w firmie prowadzącej badania rynkowe i społeczne. Od sierpnia 1996 roku jest dyrektorem działu analiz i projektów informatycznych firmy Softtutor w Warszawie.

Stowarzyszenie Polski Rynek Oprogramowania PRO zajmuje się głównie zagadnieniami ochrony praw autorskich do oprogramowania oraz problemami opodatkowania i dokumentowania sprzedaży oprogramowania.
Dla realizacji tych celów stworzono m.in. Polski Rejestr Oprogramowania i Agencję Ochrona Praw Autorskich Informatyki i Telekomunikacji. PRO bierze udział i samo organizuje spotkania, konferencje, seminaria i szkolenia.
PRO prowadzi także badania rynku oprogramowania w Polsce. Wynikom jednego z takich badań poświęcono niniejszy artykuł.

Doświadczenia w zakresie zastosowania metryk do oceny jakości oprogramowania

Janusz Górski, Marek Jurkowlaniec, Bartosz Nowicki

Jednym z zasadniczych elementów związanych z Zapewnieniem Jakości Oprogramowania jest wykorzystanie metryk (zarówno w odniesieniu do procesu wytwórczego jak i wytwarzanego produktu).

Wśród różnych działań i zadań związanych z cyklem życia oprogramowania (*software lifecycle*) Zapewnienie Jakości Oprogramowania - ZJO (*Software Quality Assurance*) cechuje szczególnie dynamiczny wzrost zainteresowania. Zapewnienie jakości jest zadaniem z zakresu zarządzania, którego głównym celem jest identyfikacja problemów związanych z jakością oraz planowanie i podejmowanie odpowiednich działań, począwszy od faz najwcześniejszych, w obrębie całego cyklu życia produktu informatycznego. Dotychczasowa praktyka w zakresie zapewnienia jakości oprogramowania dotyczy głównie końcowych etapów wytwarzania i koncentruje się przede wszystkim na identyfikacji defektów i następnie próbach naprawy działającego już programu. Podejście takie jest zdecydowanie nieadekwatne jeżeli zauważymy, że w wypadku gdy problemy jakościowe wystąpią w późnych fazach rozwoju, koszt ich usunięcia jest tak duży, że może stanowić zagrożenie dla całego projektu.

ZJO obejmuje nie tylko sam produkt informatyczny lecz również proces wytwórczy, który prowadzi do realizacji tego produktu. W zakresie produktu informatycznego ZJO ma na celu zagwarantowanie, że oprogramowanie spełnia wymagania użytkownika oraz, że oprogramowanie to może być pielęgnowane (*maintainability*) w całym okresie jego użytkowania. Z punktu widzenia procesu, ZJO koncentruje się na zdefiniowaniu pośrednich celów jakościowych (których osiągnięcie znajduje swoje odzwierciedlenie w jakości produktu) oraz określeniu i wdrożeniu efektywnych procedur, technik i narzędzi, które pozwolą na realizację tych celów.

Obiektywny pomiar stanowi podstawę wszelkiej oceny, a więc w szczególności stanowi również zasadniczy element inżynierii oprogramowania – jeżeli nie potrafimy pomierzyć tego co robimy nie mamy możliwości oceny czy osiągamy zamierzone cele oraz nie potrafimy ocenić wartości stosowanych metod i narzędzi. Drogą do oceny oprogramowania jest użycie metryk. Metryki mogą mierzyć zarówno proces rozwoju jak i różne aspekty półproduktów oraz końcowego produktu. Chociaż potrzeba użycia metryk w inżynierii oprogramowania została zidentyfikowana stosunkowo dawno, wciąż nie są one szeroko stosowane jako środek kontroli i zapewnienia jakości. Podstawowym i ciągle nierozwiązanym problemem jest trudność w zdefiniowaniu efektywnych metryk, które mierzą te atrybuty oprogramowania, które są przedmiotem bezpośredniego zainteresowania. Powoduje to powstanie luki semantycznej pomiędzy tym co rzeczywiście może zostać zmierzone (atrybuty mierzalne) a tym co powinno zostać zmierzone (atrybuty docelowe). W celu rozwiązania tego problemu przyjmuje się zwykle, że istnieje pewna (przyjęta a'priori) relacja pomiędzy atrybutami mierzalnymi i docelowymi. To założenie umożliwia wyrażenie atrybutów docelowych w terminach atrybutów mierzalnych. Walidacja tej relacji, tzn. upewnienie się, że w adekwatny sposób pozwala ona odwzorować odpowiednie atrybuty, jest problemem trudnym. W szczególności, nie jest znany wpływ takich czynników jak stopień złożoności projektu, doświadczenie zespołu, przyjęta metoda wytwarzania, itp. na postać relacji pomiędzy mierzalnymi i docelowymi atrybutami jakościowymi. Postęp zależy w znacznym stopniu od zgromadzenia i dostępności danych eksperymentalnych związanych z realizacją starannie zaplanowanych i kontrolowanych projektów w zakresie inżynierii oprogramowania. Jest to wciąż obszar aktywnych badań. Projekty studenckie są dobrym polem doświadczalnym dla tego typu badań – charakteryzują się ograniczonym rozmiarem, dobrze zdefiniowanym cyklem życia oraz dają możliwość zebrania wyników z większej liczby porównywalnych projektów. Odwołując się do teorii pomiaru, projekty pozwalają na pozyskanie doświadczalnego systemu relacji (konkretnych programów wraz z dotyczącymi ich obserwacjami), który umożliwia identyfikację oraz zrozumienie atrybutów podlegających pomiarowi.

W artykule opisano doświadczenia dotyczące użycia metryk oprogramowania do oceny programów studenckich opracowywanych w ramach Laboratorium Inżynierii Oprogramowania. Przedstawiono zakres przedsięwzięcia, jego cele, wybrane podejście, realizację, uzyskane rezultaty oraz wnioski.

Zakres i cel przedsięwzięcia

Głównym celem było opracowanie efektywnego sposobu oceny studenckich projektów realizowanych w zakresie Laboratorium Inżynierii Oprogramowania. Każdego roku sześciu nauczycieli akademickich nadzoruje wykonanie około 35 różnych projektów. Studenci pracują w grupach dwuosobowych. Zadaniem każdej grupy jest realizacja oprogramowania zgodnego z założoną specyfikacją. Wstępna specyfikacja jest celowo przekazywana studentom w formie niekompletnej i niejasnej. Grupa musi opracować precyzyjną specyfikację modułu, na podstawie tej specyfikacji zbudować plan testów, a następnie zaprojektować i zaimplementować moduł. Najczęściej tematem projektu jest generyczny abstrakcyjny typ danych, np. kolejka, zbiór, drzewo, itp. Implementacja jest wykonywana w języku C. Rozmiar modułu waha się od 500 do 700 linii kodu. Moduły są generyczne, tzn. mogą obsługiwać dowolny typ danych (kolejka, zbiór lub drzewo danych dowolnego typu). Osiągnięcie tej cechy w programie napisanym w języku C wymaga znacznego rozluźnienia reguł dotyczących zgodności typów co powoduje, że znaczna liczba błędów nie zostaje wykryta przez kompilator. Jednym z celów ZJO jest zniwelowanie tego efektu poprzez zastosowanie innych technik pro jakościowych.

Użycie metryk miało następujące cele:

- stworzenie podstaw do jednorodnej oceny projektów,
- zmniejszenie wysiłku koniecznego do oceny prac studenckich,
- dostarczenie studentom środków umożliwiających pomiar jakości w czasie tworzenia oprogramowania.

Zastosowane podejście

Realizacja przedsięwzięcia wymagała rozwiązania następujących problemów:

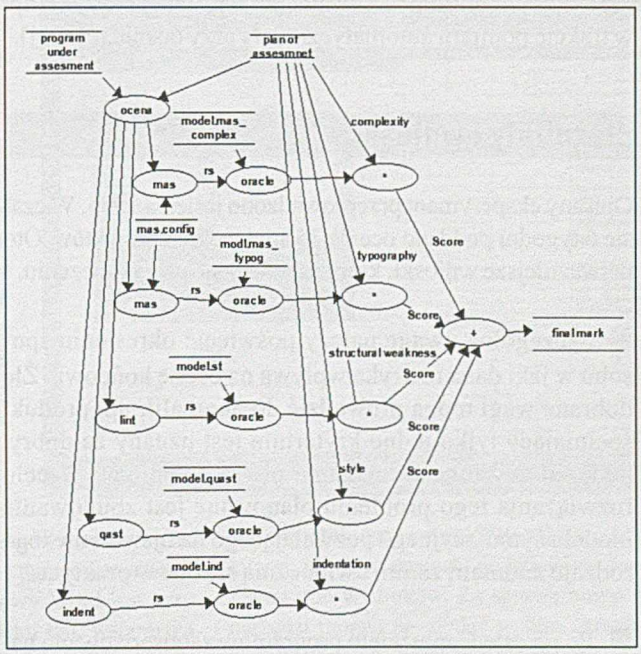
- jakie charakterystyki oprogramowania (atrybuty docelowe) podlegają ocenie,
- jakie atrybuty podlegają bezpośredniemu pomiarowi,
- dobór metryk dla atrybutów mierzalnych,
- w jaki sposób wywieść z przeprowadzonych pomiarów końcową ocenę wyrażoną w terminach wartości atrybutów docelowych,
- jak zapewnić spójność pomiędzy metrykami a standardami projektowania i kodowania,
- w jakim stopniu pomiary mogą być wspomagane przez narzędzia.

Analiza charakterystyki projektów studenckich doprowadziła do wyodrębnienia zbioru atrybutów docelowych. Z każdym atrybutem docelowym związano zbiór metryk określających wartości atrybutów mierzalnych:

- *złożoność funkcji*
  - * złożoność cyklomatyczna
  - * poziom zagnieżdżeń strukturalnych
- *topografia funkcji*
  - * liczba linii kodu
  - * procentowy udział zmiennych ze znaczącymi nazwami
  - * procentowy udział komentarza w funkcji
  - * obecność nagłówka
  - * procentowy udział złożonych wyrażeń
- *styl projektu funkcji*
  - * pojawienie się więcej niż jednej instrukcji „return"
  - * użycie „goto", „continue", „break" i operatora przecinka
  - * użycie stałych numerycznych zamiast stałych symbolicznych
  - * wystąpienie pętli z niezdefiniowanym warunkiem stopu
- ***błędy konstrukcyjne.*** Pokrywające między innymi:
  - * niezainicjalizowane zmienne
  - * niewykorzystane zmienne i funkcje
  - * naruszenie reguł związanych z konwersją typów
- *struktura kodu*
  - * zgodność ze standardem wcięć

Realizacja

W realizacji prezentowanego podejścia zastosowano kilka istniejących, publicznie dostępnych narzędzi. Metryki związane ze złożonością i topografią funkcji są dostępne w narzędziu MAS (Maintainabilty Assesmnets Tools - narzędzia opracowane na Uniwersytecie Idaho w porozumieniu z firmą Hewlett Packard [8]). Styl programu jest oceniany przez QU-



Rys. 1. Diagram przepływu danych dla procesu oceny.

AST - program opracowany w ramach pracy dyplomowej w EFP [7]. Program LINT [9], w który wyposażony jest standardowo system operacyjny UNIX, wspomaga ocenę błędów projektowych. Praca w tak urozmaiconym środowisku wymaga pewnej wspólnej platformy służącej jako baza do integracji. CEILIDH [1] (Computer Environment for Interactive Learning in Divers Habitats) jest środowiskiem opracowanym na Uniwersytecie Nottingham, które oprócz innych usług, oferuje mechanizm pozwalający na łączne użycie różnych metryk. Program ORACLE [2], będący częścią systemu CEILIDH, wraz z skryptami systemu UNIX i plikami konfiguracyjnymi stanowi narzędzie pozwalające na przetwarzanie wyników uzyskanych z zastosowania wielu metryk.

Diagram przepływu danych z Rysunku 1 przedstawia proces oceny programu realizowany w ramach środowiska CEILIDH.

Ocena projektów (rozróżnienie pomiędzy dobrymi i złymi) wymagała zdefiniowania punktu odniesienia. W tym celu wprowadzono standard [3] określający te wymagania jakościowe, które były uwzględnione w procesie oceny. Standard ten stanowił swego rodzaju kontrakt pomiędzy zespołem projektującym a zespołem oceniającym.

Metryki zostały opisane zgodnie z następującą strukturą:

nazwa metryki
semantyka - dokładne wyjaśnienie co dana metryka mierzy
kryterium - wartości określające zakres akceptowalności pomiaru
odniesienie od standardu - dyskusja czy, gdzie i w jakim stopniu metryka jest uwzględniona w standardzie kodowania
ocena - sposób, w który metryka wpływa na końcową ocenę.

W procesie oceny część metryk jest obliczana ze wspomaganiem narzędziowym a pozostałe są obliczane „ręcznie". Zakres oceny ręcznej został ściśle określony w postaci formularzy kontrolnych. Dzięki temu uniknięto niebezpieczeństwa wielokrotnego uwzględnienia danej metryki (np. w trakcie pomiaru automatycznego i przy ocenie ręcznej).

Rezultaty i wnioski

Opisany eksperyment przeprowadzono jesienią 1995. W czasie 6 tygodni poddano ocenie 35 studenckich projektów. Oto najważniejsze wnioski, które nasunęły się po zakończeniu.

■ Szczególną uwagę należy poświęcić określeniu sposobu w jaki dana metryka wpływa na ocenę końcową. Źle dobrane wagi mogą prowadzić do anomalii, np. produkt spełniający tylko jedno kryterium jest uznany za dobry, podczas gdy inne wymagania nie są spełnione. W celu rozwiązania tego problemu planowane jest zbudowanie modelu symulacyjnego pozwalającego na ujawnienie tego rodzaju anomalii zanim uwidocznią się one w praktyce.

■ Po prześledzeniu wielu programów zaobserwowano następującą zależność: dobry program (oceniony jako dobry przez pewną grupę metryk) jest również oceniony jako dobry przez inne metryki, pomimo iż pozornie zakres tych metryk jest znacząco inny. Sugeruje to możliwość zmniejszenia zbioru metryk wystarczających do wyłonienia tych programów, które są złe. Należałoby również wykluczyć sytuację, w której studenci dostosowują się do konkretnego zbioru metryk, przestrzegając tylko tych wymogów, które podlegają bezpośredniej ocenie. Rozwiązaniem może tu być stosowanie metryk losowanych ze zbioru podstawowego.

■ Ważym efektem przedsięwzięcia jest kalibracja relacji zastosowanej dla odwzorowania pomiarów w atrybuty docelowe. Zebrane dotychczas dane eksperymentalne wykorzystano do doboru wag określających wpływ poszczególnych pomiarów na ostateczną ocenę jakościową, co pozwala domniemywać, że w ramach następnej tury eksperymentów ocena końcowa będzie bardziej wiarygodna. Dobrze dostrojone środowisko oceny pozwali na zwiększenie obiektywizmu oceny programów oraz oszczędzi czas poświęcony na ocenę.

■ Zdobyte doświadczenia wskazują na przewagę metryk prostych. Są one intuicyjnie zrozumiałe oraz są efektywnie realizowalne.

■ Wśród planów na przyszłość jest zastosowanie metryk do kontroli przebiegu całego projektu, a nie tylko produktu końcowego oraz opracowanie zbioru metryk działających na poziomie całego programu (a nie pojedynczego modułu). Wybrany zbiór metryk zostanie zaimplementowany w wielu wersjach pozwalających na badanie jakości kodu w różnych językach.

Literatura

[1] Beford S., Burke E., Foxley E., Gutteridge N., Zin A.M.: ***The CEILIDH System***, Departament of Computer Sciences, University of Nottingham, UK, 1994
[2] Zin A.M., Foxley E.: ***The „oracle" program***, Departament of Computer Sciences, University of Nottingham, UK, 1994
[3] Jurkowlaniec M.: ***Laboratorium Inżynierii Oprogramowania: Standard Kodowania w języku C***, EFP, 1995
[4] Fenton N.E.: ***Software Metrics. A Rigrorous Approach***, Chapman& Hall, 1991
[5] Shepperd M.J.: ***Design metrics: an empirical analysis***, Software Eng J 5(1),1990
[6] Boehm B.W.: ***Characteristics of software quality***, North Holland, 1978
[7] Stańczak J.: ***Narzędzie do oceny jakości oprogramowania***, praca magisterska pod kierunkiem prof. J. Górskiego, EFP, 1995
[8] Anderson C., Hall K., Lowe V., Rhoades T., Steed D.: ***Maintainability Assesment System for C version 4.0. User's Manual***, 1992
[9] Dokumentacja techniczna systemu BOSX.

Artykuł ten był prezentowany na II Konferencji „Informatyka na Wyższych Uczelniach dla Gospodarki Narodowej".

Janusz Górski, Marek Jurkowlaniec i Bartosz Nowicki są pracownikami Instytutu Technik Telekomunikacyjnych i Informatycznych w Poznaniu.

Konsekwencja w każdym calu

Microsoft Office 97 po polsku

Tomasz Szewczyk

W niecałe trzy miesiące po amerykańskiej premierze, a więc zgodnie z zapowiedziami, pojawił się na naszym rynku w pełni spolonizowany następca najpopularniejszego w świecie pakietu zintegrowanych aplikacji biurowych - Microsoft Office 97 dla Windows 95 i NT. Ten nowoopracowany zbiór znanych i lubianych programów znakomicie sprawdza się w szeroko rozumianych zastosowaniach biznesowych, a zmiany wprowadzone do najnowszej wersji, będące wynikiem ponad 25 tys. godzin badań nad jego funkcjonalnością powinny sprawić, iż okaże się on jeszcze łatwiejszy i przyjaźniejszy w użyciu.

Co i jak?

Polskojęzyczna edycja najnowszej wersji MS Office została po raz pierwszy zaprezentowana publicznie podczas niedawnych targów Infosystem '97. Produkt jest zatem „jeszcze gorący", a pamiętać trzeba, iż jako jedyny na naszym rynku oferuje komplet w pełni 32-bitowych programów, które komunikują się z użytkownikami w naszym rodzimym języku. W handlu Microsoft Office 97 dostępny ma być tradycyjnie w dwóch wersjach - standard i professional. W pierwszej z nich pakiet składać się będzie z: Word'a, Excel'a, Powier Point'a oraz aplikacji wspomagającej zarządzanie informacjami biurowymi o nazwie MS Outlook. W drugiej zaś, prócz wymienionych odnajdziemy w nim ponadto system zarządzania bazami danych MS Access 97.

Przejdźmy na chwilę do spraw technicznych, pakiet Microsoft Office przeznaczny jest wyłącznie dla systemów operacyjnych Windows 95 i NT, do poprawnej pracy wymaga komputera klasy 486 (lub lepszego) oraz minimum 12 MB pamięci w przypadku Windows 95 lub 16 MB w przypadku NT Workstation. Pakiet ten na dysku twardym zależnie od wybranego sposobu instalacji zajmuje 73 do 191 MB.

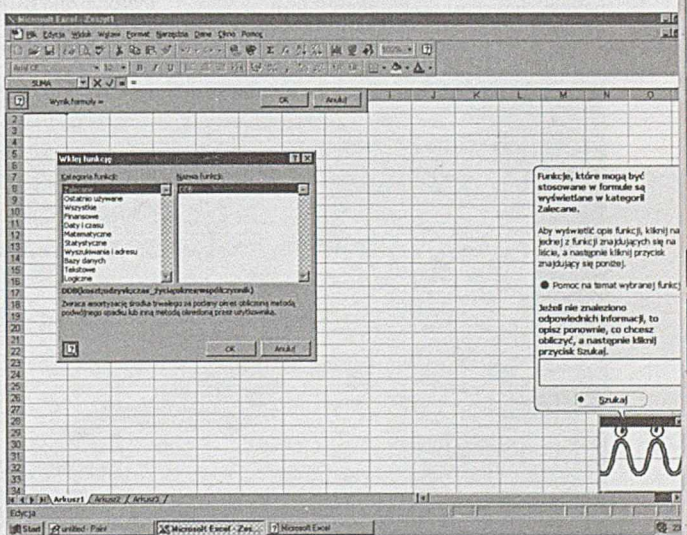
Word

O tym jak wielkie zmiany zaszły w aplikacjach nowego pakietu najlepiej przekonuje nas praca z MS Word 97. Program ten potrafi m.in. znacznie sprawniej wykrywać błędy ortograficzne i literówki oraz podpowiadać nam poprawną pisownię wyrazów i synonimy bezpośrednio w czasie pisania. Ponadto za sprawą, znanej już z wersji 6, a obecnie udoskonalonej technologii Intellisense, objawiającej się m.in. w takich udogodnieniach jak Autokorekta, potrafi wstawiać specyficzne symbole i znaki, wykrywać przypadkowe wciśnięcia CapsLock'a oraz automatycznie rozpoznaje sytuacje, w których użytkownik zamierza tego typu poprawki wprowadzane przez program zignorować. Do tego dochodzą jeszcze „inteligentne" autowybieranie wyrazów oraz fragmentów tekstu i tak „standardowe" narzędzia jak szybkie formatowanie, autoformatowanie, malarz formatów, możliwość wstawiania znaków wodnych i innych efektów do tekstu oraz mnóstwo kreatorów i szablonów, które potrafią wykonać wiele kłopotliwych czynności, takich jak np. pisanie listów, wysyłanie faksów, czy tworzenie życiorysów prowadząc za użytkownika niemal krok po kroku.

Godnymi odnotowania cechami MS Word 97 są także łatwa możliwość wykorzystania tego programu jako edytora poczty elektronicznej, przystosowanie go do roli edytora i przeglądarki internetowych dokumentów tzn. do pracy w formacie HTML oraz niemal doskonała integracja z innymi narzędziami Office tzn. np. z Power Point, dla którego program ten tworzy konspekty prezentacji, Excel'em który potrafi udostępniać swoje arkusze i wykresy dokumentom tekstowym, czy Access'em i Outlookiem, z których to programów możemy m.in. pobierać adresy w czasie generowania korespondencji seryjnej. Nowy Word jest przy tym całkowicie po polsku tzn. wspomaga użytkownika nie tylko polskim słownikiem ortograficznym, czy polskim słownikiem wyrazów bliskoznacznych, lecz także przesympatycznym asystentem, domyślającym się dość inteligentnie co w danej chwili chcemy zrobić. Program ten dysponuje także wieloma spolszczonymi szablonami, stylami i kreatorami oraz kilkoma krojami czcionek ozdobnych, które posiadają polskie znaki. Nadaje się też świetnie do redagowania długich i skomplikowanych dokumentów, gdyż można go ustawić w taki tryb podglądu, że z lewej strony będziemy widzieć konspekt dokumentu, a z prawej jego treść, że nie wspomnę już o takich udogodnieniach jak np. możliwość kreślenia układu tabeli myszą, czy tradycyjna już dla tego edytora łatwość i elastyczność stosowania tzw. stylów.

Excel

Excel 97 jest nowy i szybszy w pracy, można by powiedzieć od podstaw. Program ten został bowiem wyposażony w inny od dotychczas stosowanego mechanizm edycji formuł - pozwala np. pisać wyrażenia bez konieczności odwoływania się do tradycyjnych odnośników rodem z gry w okręty (np.



A1, czy B4), oznacza zakresy różnymi kolorami, itd. oraz pozwala na łatwiejsze formatowanie wydruków. Za sprawą nowych narzędzi można np. dodać w nim „koszty kwietnia" do „kosztów maja", czy też zsumować „wydatki na energię" posługując się

wyłącznie nazwami nagłówków kolumn i wierszy tabeli, (nie trzeba chyba nikogo przekonywać o tym, że łatwiej jest w tak zapisanych wyrażeniach odnaleźć też ewentualne błędy). Znacznie też łatwiej drukować duże tabele w taki sposób, by nie trzeba było potem ich ciąć i sklejać, nie mówiąc już o samej estetyce wydruków.

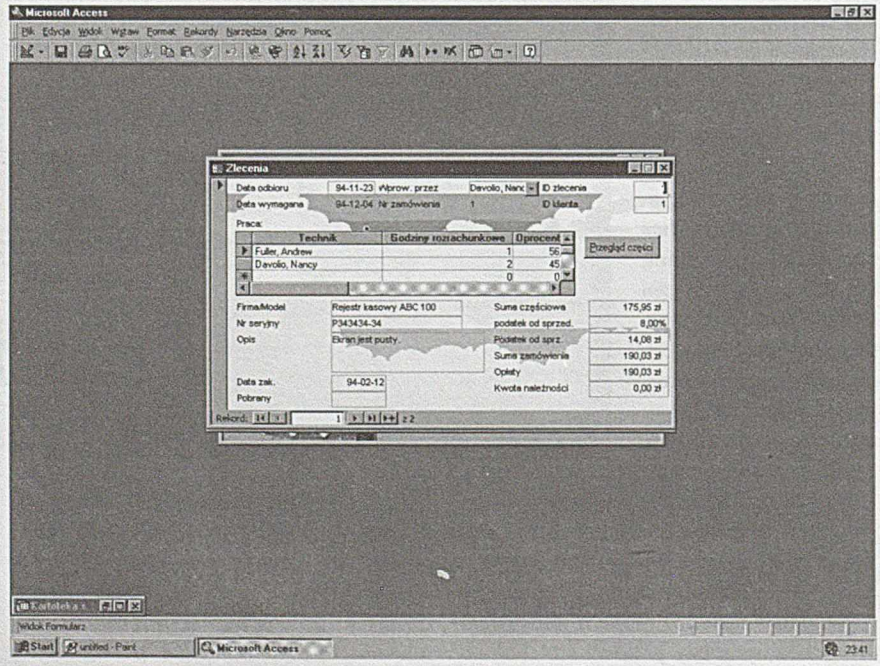
Nowy Excel ma prócz wymienionych inną, nową funkcję - tzw. Autokorektę wyrażeń tzn. sam dodaje pominięte nawiasy i inne znaki podczas pisania formuł. Udostępnia też Autowypełnianie, za sprawą którego potrafi podpowiadać użytkownikowi takie wartości jak kolejne dni tygodnia, nazwy miesięcy, itd. oraz udostępniać wartości tekstowe, które zostały wpisane w bezpośrednim sąsiedztwie aktualnie edytowanej komórki. W programie tym ułatwione jest ponadto poruszanie się po arkuszu za sprawą wskazówek pojawiających się na belkach przewijania, kontrolowanie importu plików tekstowych (funkcja „Zamień tekst na kolumny") oraz wydawanie poleceń takich jak np. „Wstaw kolumnę" - gdyż są one obecnie wykonywane z wykorzystaniem animacji, a jeśli trzeba to i Asystenta.

Ciekawostką Excel'a jest także wprowadzona w wersji 95 funkcja filtrowania „10 - naj" oraz Autokalkulacja. Za sprawą tej ostatniej wystarczy zaznaczyć blok komórek, żeby w pasku stanu obejrzeć sumę ich zawartości. W programie tym odnajdziemy też - „Data Map", służącej do wizualnej analizy informacji poprzez tworzenie map, na których dane będą zlokalizowane graficznie i oczywiście wszechobecne narzędzia internetowe łącznie z możliwością zapisu arkusza w formacie HTML i wpisywania odwołań inter- i intranetowych wprost do komórek i formuł.

Do tego dochodzą jeszcze, tak jak w przypadku Word'a, szablony i narzędzia językowe, które są wspólne dla całego pakietu Office, akceptowanie formularzy bazy danych Access (tłumaczenie ich na arkusze kalkulacyjne i odwrotnie) oraz możliwość takiego oprogramowania arkuszy za pomocą wbudowanego języka makroprogramów (VBA), by były one „jeszcze przydatniejsze" dla użytkownika. Szczególnie, że Microsoft dostarcza nam od pewnego czasu z programem takie rodzynków jak: Data Access Objects (możliwość wykorzystania przez arkusz operacji swoistych dla bazy Access), Improved Object Model (kontrola nad obiektami i nowymi możliwościami OLE), czy też wspomniany Visual Basic for Applications, a więc „gotowiec" do budowania kodu programu niemal z klocków.

Power Point

Program wspomagający przygotowywanie pokazów multimedialnych - Power Point na pozór niewiele różni się od swych poprzednich wersji, choć został napisany niemal całkiem od nowa i działa zauważalnie szybciej niż poprzednik. Użytkownik ma tu do dyspozycji ponad 30 gotowych szablonów prezentacji, może także tworzyć własne, korzystając z galerii własnych lub dostarczonych przez producenta rysunków, dźwięków oraz bogatych efektów animacyjnych. Podczas pracy z Power Point istnieje możliwość zaimportowania konspektu z Word'a oraz możliwość przesyłania slajdów do Word'a w celu zilustrowania tworzonych dokumentów. Można także gotowe prezentacje publikować w Internecie, co stwarza, przynajmniej teoretycznie bardzo ciekawe możliwości.

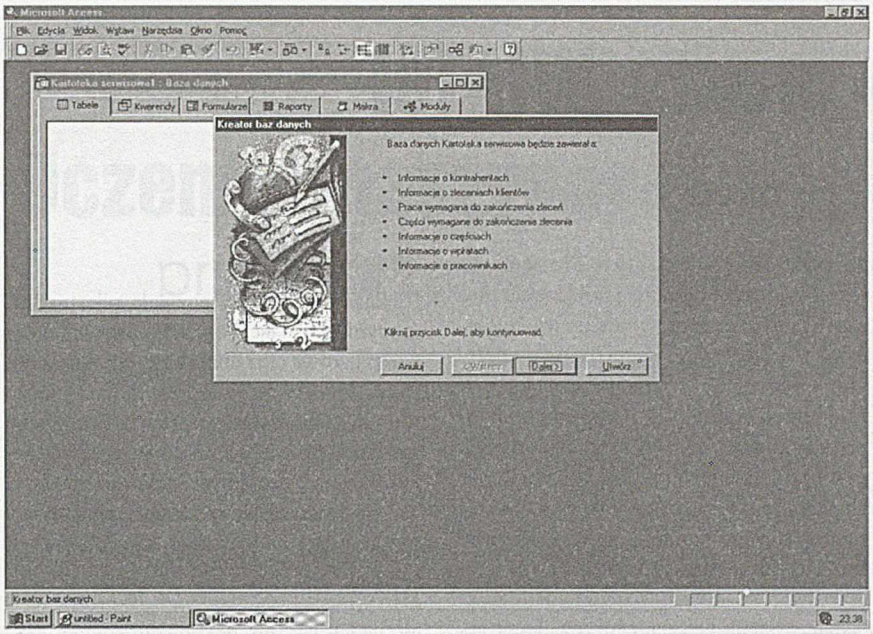


Każda prezentacja Powier Point'a, która jest zapisane w jednym pliku dyskowym może być odtwarzana w kilku wariantach czasowych, tzn. prezentacja może obejmować dowolną liczbę slajdów znajdujących się w danym pliku. Mamy tu też do dyspozycji mechanizm pozwalający na dołączanie slajdów pochodzących z innych animacji, przy czym program automatycznie dostosowuje tło i inne atrybuty do tego, co „mamy na warsztacie". Podobnie wiele udogodnień przewidziano dla osób prowadzących pokaz - możemy myszką zakreślać fragmenty ekranu, wbudowywać w slajdy przyciski wywołujące zadane działania, a prezenter ma możliwość robienia w komputerze notatek, które są dla publiczności niewidoczne. Autorzy wbudowali też w program wiele ciekawych narzędzi, takich jak np. automatyczny dzielnik i podsumowywacz slajdów, który z jednego zbyt długiego slajdu zrobi za nas kilka szczegółowych lub odwrotnie, czy też niezwykle przydatny autokorektor. Wszak nie ma chyba większego wstydu niż wykryta przez publiczność literówka lub co gorsza błąd ortograficzny.

Outlook

MS Outlook 97 jest nową aplikacją w rodzinie MS Office. Program ten łączy w sobie zalety programu do obsługi poczty elektronicznej z obsługą osobistego kalendarza, harmonogramu grupy, terminarza, bazy adresowej, notatnika i zarządcy dokumentów, a z technicznego punktu widzenia stanowi kontynuację dotychczasowych rozwiązań, takich jak Schedule+, MS Mail i Exchange. Outlook to program dziwny na pierwszy rzut oka. Specjaliści z Microsoftu twierdzą jednak, że dzisiejsi użytkownicy komputerów często cierpią nie na brak, lecz na nadmiar informacji i że brakuje im jedynie narzędzi do ich wyszukiwania i przechowywania. Trudno powiedzieć czy mają rację, z poziomu Outlook jednakowoż możemy np. aranżować spotkania w grupie i zapisywać ich terminy w prywatnym lub grupowym kalendarzu, zarządzać dokumentami i korespondencją znajdującą się w różnych miejscach sieci, zanotować uwagi na tzw. „żółtych kartkach", czy też, posługując się wbudowanymi formularzami elektronicznymi przeprowadzić głosowanie na wybrany temat.

Nie koniec na tym, program ten rozpoznaje np. co to znaczy „następny czwartek" (umie wyliczyć kiedy to będzie i wstawić do listu), podzieli za nas napisany ciurkiem adres na poszczególne pola (imię, nazwisko, ulica, itd.), a po przeciągnięciu adresu na ikonę korespondencji sam zaadresuje list. Takich udogodnień w Outlook'u jest znacznie więcej, a warto pamiętać, że do dyspozycji naszej został oddany także Dziennik - automatyczny rejestr zdarzeń, który pozwala



nam łatwo odnaleźć np. arkusz, z którym pracowaliśmy np. w ubiegły piątek.

Access

Opisywanie systemów zarządzania bazami danych to temat niewdzięczny - posługiwanie się nimi jest zazwyczaj trudne, a samodzielne stworzenie najprostszej aplikacji, takiej jak chociażby książka adresowa przekracza z reguły zdolności nie związanego zawodowo z programowaniem użytkownika. Z drugiej jednak strony systemy te są najpowszechniej bodaj stosowanymi narzędziami służącymi do tworzenia aplikacji gospodarczych - w zasadzie każdy system służący do obsługi firmy musi z któregoś z nich korzystać.

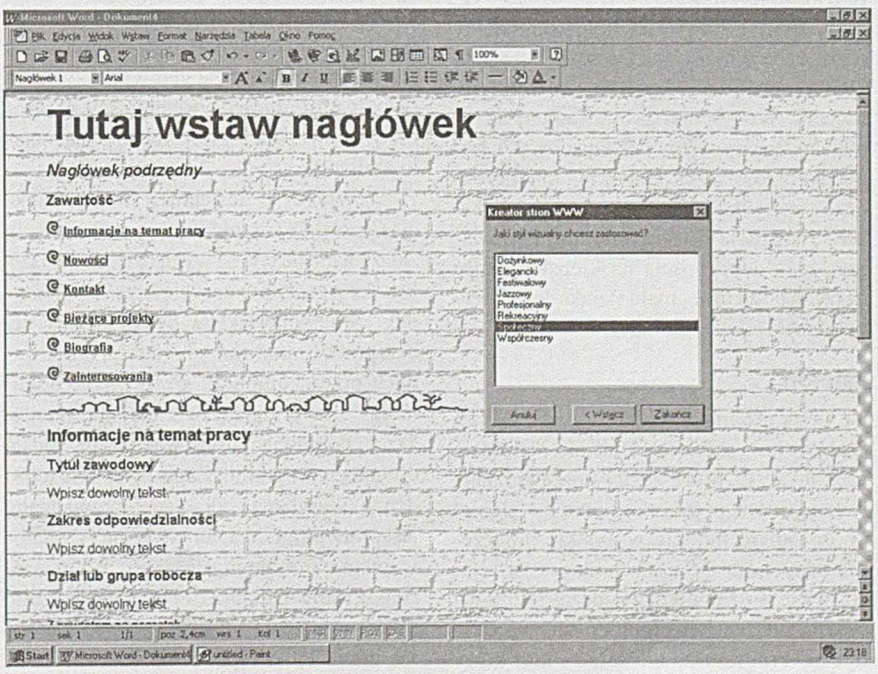
Microsoft Access jest programem, który ma do zaoferowania „coś użytecznego" zarówno dla zwykłych użytkowników, jak i dla profesjonalistów. Ci pierwsi odnajdą w nim bowiem narzędzia wspomagające ich pracę podczas tworzenia własnych, prostych rozwiązań w postaci kreatorów baz danych, kwerend, tabel i raportów. Drudzy zaś otrzymają wygodne środowisko programistyczne, które uzupełnione o pewne dodatkowe elementy pozwala na efektywne opracowywanie bardziej złożonych projektów.

O tym jak wiele udało się osiągnąć programistom Microsoftu w tym względzie może się przekonać nawet całkiem nieobeznany z bazami danych użytkownik, który w ciągu kilkunastu minut jest w stanie stworzyć jedną z ponad 20 rodzajów relacyjnych baz danych, takich jak np. rejestr zamówień, kartoteka osobowa, rejestr środków trwałych czy baza danych kontrahentów. Niezaprzeczalną zaletą obecnej edycji Access'a jest też możliwość łatwej publikacji danych w Internecie oraz korzystania z odległych baz we własnych formularzach. Cechy te pozwalają na zdalne odświeżanie danych w bazie przez wielu użytkowników oraz na synchronizację tych zmian przez Internet, co może się przydać np. w sytuacji, gdy firma ma wiele oddziałów lub współpracowników rozsianych na dużym obszarze. Nie mówiąc już o tym, że można sobie w ten sposób ułatwić choćby obsługę zamówień pochodzących od różnych kontrahentów.

Nowy styl pracy

Wydaje się, że programiści z Microsoftu tworząc nowy pakiet koncentrowali się głównie na tym, by użytkownik ich oprogramowania pracował nie tyle z komputerem ile z dokumentem i znanymi mu narzędziami oraz był w maksymalnym stopniu wspomagany podczas pracy przez inteligentnie reagujący na jego potrzeby system pomocy, uzupełniony gdzie się da automatami pozwalającymi wykonywać mu najbardziej typowe czynności. Udało się im to osiągnąć w dużym stopniu, czemu sprzyja zarówno architektura całego pakietu - aplikacje tworzące go mają znaczną część wspólnego kodu źródłowego, przez co stały się bardziej do siebie podobne w działaniu i wyglądzie. Sprzyja także i to, że wszystkie programy łatwiej niż dotychczas potrafią ze sobą wymieniać informacje.

W tym miejscu na wyróżnienie zasługuje nie tylko możliwość szerszego niż dotychczas korzystania z mechanizmów takich jak OLE czy DDE w poszczególnych elementach pakietu. Wyróżnić także należy np. Spinacz Office, za sprawą którego możemy stworzyć dokument składający się z niemal dowolnej ilości tekstów, arkuszy, prezentacji, baz danych, itd., a następnie edytować go lub drukować jako całość i przechowywać w jednym pliku dyskowym. Z tego punktu widzenia niebanalną zaletę tego akurat oprogramowania stanowi także bardzo dopracowana integracja funkcjonalna Word'a z Excel'em i Power Pointem, Excel'a z Accessem, czy też Outlook'a z pozostałymi aplikacjami oraz wszystkich wymienionych aplikacji z Internetem, co stwarza użytkownikom całkowicie nowe możliwości organizowania sobie pracy oraz wymiany dokumentów. Reasumując, nowy Office nadaje się równie dobrze do klasycznej pracy, kiedy to całość działań użytkownika zaczyna się i kończy na pojedynczym komputerze, do pracy w grupie działającej lokalnie lub w sieci rozległej oraz do zastosowań, w których źródłem będą dane pochodzące z dużych, wykorzystujących superkomputery systemów, a efekty będą zapisywane w zegarkach-notesach takich jak Timex Data Watch.



Po prostu WinTel

Tomasz Szewczyk

Z prawdziwą przyjemnością prezentujemy dziś rdzennie polskie rozwiązanie komunikacyjne, które przewyższa swą funkcjonalnością większość dostępnych na naszym rynku rozwiązań importowanych. Produkt ten nosi nazwę WinTel, jest przeznaczony dla komputerów PC, a składa się z modemu oraz oprogramowania, które prócz funkcji telefonicznych, faksowych i modemowych pozwala na przekazywanie obrazów na odległość.

Miłe początki

Podstawę pakietu WinTel stanowi program komunikacyjny, który jest przeznaczony do pracy w środowisku Windows (3.x lub 95). Program ten składa się z kilku części: Telefon, Faks, Modem, Video i Rozpoznawanie mowy (opcja), które są połączone funkcjonalnie w jedną spójną całość. W aplikacji tej przewidziano też możliwość wydawania komputerowi poleceń głosem oraz funkcje automatycznej sekretarki, która działa podobnie jak w aparatach telefonicznych, z tym, że program analizuje treść rejestrowanych nagrań.

Aplikacja ta potrafi obsługiwać modemy z funkcją Voice, warto więc zwrócić uwagę, że choć urządzenia takie wytwarza wielu producentów (np. Commstar, Microcom, Zoltrix, Zoom czy Zyxel), firma ComPol II, jako jedyny krajowy producent, oferuje własne rozwiązania hardware'owe tego typu.

W ofercie firmy znajdują się modemy o różnych szybkościach transmisji - 14 400, 28 800 oraz 33 600 bps - zależnie od ceny i wersji. Wraz z oprogramowaniem otrzymaliśmy ten ostatni model, wykonany w formie karty rozszerzeń. On też był używany w czasie naszych redakcyjnych testów i wykazał się łatwością instalacji oraz konfigurowania, dużą odpornością na kiepską jakość naszych łącz i inne „mało standardowe" wybryki sieci telekomunikacyjnej.

Telefon

Ta opcja programu WinTel odpowiada za głosowe połączenia telefoniczne. Należą do nich także te, które zostały przewidziane w ustalonym przez użytkownika planie. W opisanym module wszystkie połączenia przychodzące i wychodzące są rejestrowane w dzienniku, a podczas rozmowy można nagrać ich przebieg na twardym dysku. Program zawiera też Plan, który gromadzi informacje o tym, kto zamawia rozmowę, dzień i godzinę połączenia oraz nazwisko lub numer abonenta. Można zatem obowiązek pamiętania o części telefonów powierzyć po prostu komputerowi.

Ciekawe, że gdy korzystamy z WinTel'a, po uzyskaniu połączenia komputer sam inicjuje rozmowę przygotowanym wcześniej komunikatem głosowym. Informuje też zamawiającego o nawiązaniu połączenia. Można więc po przyjściu do pracy „zaprogramować" rozmowę z abonentem, który będzie osiągalny wyłącznie o zadanej godzinie. Komputer w odpowiednim czasie sam wybierze numer (będzie go wybierał do skutku), a po nawiązaniu połączenia zakomunikuje rozmówcy „łączę rozmowę z..." i dopiero wtedy poinformuje nas o nawiązaniu połączenia dzwonkiem.

Faks

Moduł ten nie różni się zbytnio od podobnych, dostępnych w zagranicznych programach, które mamy na rynku. Wspomaga on przygotowywanie, wysyłanie, odbieranie i zarządzanie całą korespondencją faksową oraz umożliwia planowanie automatycznego wysyłania dokumentów do jednego lub grupy odbiorców w okresie najtańszych taryf telefonicznych. W czasie przygotowywania treści faksu można korzystać z dowolnego edytora tekstów pracującego w środowisku Windows. Użytkownik może ponadto samodzielnie zaprojektować dowolnie wiele stron tytułowych dla faksów wychodzących wykorzystując obiekty OLE. Łatwo więc umieścić na stronie tytułowej np. logo firmy, czy inne znaki graficzne.

W WinTel'u dostępna jest także funkcja Quick Fax (szybki faks), przydatna, gdy zależy nam nie tyle na wyglądzie, co na możliwości wysłania czegoś „ od ręki". Wysyłanie i odbieranie faksów odbywa się w tle, bez przerywania innych zadań realizowanych przez komputer, a otrzymane dokumenty można oglądać na ekranie, powiększać, poprawiać, drukować lub przenosić do innego programu. Zapis całej korespondencji (wysyłanej i odbieranej) prowadzony jest przez program automatycznie - z zachowaniem chronologii, informacji o jej kierunku, dniu nadania/odebrania, godzinie przeprowadzenia tej operacji oraz nazwie lub numerze korespondenta.

Wideo

Ciekawe, że WinTel pozwala rejestrować, modyfikować i łączyć w sekwencje animacje komputerowe oraz obraz otrzymywany z dowolnej karty wideo, która jest zgodna z Video for Win. Obraz wprowadzony do programu jest kompresowany i może być następnie automatycznie odebrany na innym komputerze. Transmitowane pliki zachowują standardowy format *.AVI umożliwiający łączenie obrazu, dźwięku i tekstu. Dzięki zastosowaniu OLE pliki te mogą być dołączane do innych aplikacji, takich jak np. dokumenty edytora MS-Word. W programie dostępna jest też funkcja zdalnego pobierania obrazu z kamery i odsyłania go pod wskazany numer, co w pewnym ograniczonym zakresie umożliwia np. obserwację pomieszczeń. Warto też wiedzieć, że generując odpowiedni kod DTMF (tonowo) można z odległego aparatu telefonicznego sterować niektórymi funkcjami programu, co pozwala np. zlecić przesyłanie obrazu pod wskazany numer.

Rozpoznawanie mowy

Wspomniałem, że opisywany dzisiaj program realizuje ideę wykorzystania telefonu do wydawania komputerowi poleceń za pomocą głosu. Rozpoznawanie mowy działa w ten sposób, że komputer porównuje wypowiadane przez nas słowa z zapamiętanym wzorcem a następnie realizuje dowolne, przypisane im wcześniej przez użytkownika funkcje.

Funkcjami takimi mogą być np. zadania związane z obsługą systemu operacyjnego, takie jak np. uruchomienie programu czy też skasowanie pliku. Wypowiadane słowa są nagrywane i odtwarzane jako pliki dźwiękowe, a ich zapisywania i odtwarzania dokonuje się przez aparat telefoniczny i dodatkowy głośnik. Skuteczność rozpoznawania słów wynosi 70-80% i zależy od stopnia podobieństwa do siebie słów stanowiących słownik programu.

Modem

To już ostatni moduł WinTel'a, który został nam do omówienia. Umożliwia on wysyłanie i odbieranie plików z wykorzystaniem wszystkich popularnych protokołów (m.in. ZMODEM, KERMIT, XMODEM) z tym, że transmisja może mieć miejsce także równolegle z rozmową telefoniczną (full duplex) lub też być wykonywana automatycznie, w zadanym czasie.

Uczenie maszynowe – obiecującą strategią przetwarzania informacji w biznesie?

Część II

Zdzisław S. Hippe

Metodologia GTS + MLC w poszukiwaniu regularności w danych

Przed opisaniem wybranych szczegółów algorytmu GTS, zdefiniujmy kluczowe wymagania w odniesieniu do danych, w których poszukuje się regularności. Wymagania te są następujące:

- *opis pary atrybut-wartość*: analizowane dane, zawierające informacje o danym przypadku powinny być wyrażone w postaci ustalonej kolekcji atrybutów (cech). Każdy atrybut może mieć wartości w dziedzinie wartości symbolicznych (dyskretnych) lub numerycznych (ciągłych). Atrybuty porządkowe są przez aktualną wersję algorytmu traktowane jako symboliczne,
- *przestrzeń atrybut-wartość*: zakłada się, że przypadki o zbliżonych (podobnych) właściwościach tworzą skupiska w wielowymiarowej przestrzeni obrazów. Przestrzeń ta ma tyle wymiarów, ile atrybutów było użytych do opisu przypadków. Każdy atrybut musi mieć zdefiniowane (znane) wartości, ponieważ aktualna wersja algorytmu nie akceptuje „brakujących" wartości,
- *pre-definiowane klasy*: kategorie do których przypisane są rozpatrywane przypadki (binarne lub wielokategoryjne) winny być á priori znane. W terminologii uczenia maszynowego mamy tutaj zatem do czynienia z uczeniem nadzorowanym,
- *rozmyto-dyskretne klasy*: w początkowej fazie działania algorytmu, gdy główną funkcję spełnia algorytm GTS, przypadki muszą należeć do ostro zadanych klas; przypadek albo należy, albo nie należy do określonej klasy. Później, w drugim etapie poszukiwania regularności, przynależność danego przypadku do pewnej klasy nie musi być tak ściśle zdefiniowana,
- *wiarygodność utworzonego modelu klasyfikacji*: zależy od stosunku liczby przypadków w zbiorze treningowym do liczby atrybutów. Zwykle, powinno być wielokrotnie więcej przypadków od atrybutów użytych do ich opisu.

Dalsze badania nad budową i zastosowaniem narzędzi do tworzenia modeli klasyfikacji mogą spowodować konieczność dokonania pewnych zmian niektórych z wymienionych założeń.

Walidacja reguł wnioskowania w algorytmie GTS

Gdy GTS poszukuje regularności w danych i konstruuje model klasyfikacji, kolejne reguły wnioskowania są oceniane przy pomocy następującej miary:

$$H = G + \sqrt{A}$$

G: jest sumą liczby przykładów poprawnie oraz błędnie sklasyfikowanych przez tworzoną regułę (lub wyrażenie logiczne), podzieloną przez liczbę wszystkich przykładów w zbiorze treningowym,

A: jest udziałem poprawnie sklasyfikowanych przykładów (przez rozpatrywaną regułę) w zbiorze przykładów poprawnie i błędnie sklasyfikowanych.

Pierwszy człon (G) wyrażenia określa ogólność reguły, natomiast człon drugi (A), jej dokładność. Człon A, którego koncepcja jest oparta na pojęciu udziału molowego, miary powszechnie stosowanej w naukach przyrodniczych, odgrywa ***krytyczną*** rolę w procesie tworzenia reguły: rozwój reguły wnioskowania trwa aż do chwili gdy A = 1. Jest oczywistym, że wtedy reguła odnosi się tylko do jednej klasy. Stwierdzono, że miara H przewyższa pod wieloma względami inne metody oceny reguł wnioskowania, zwłaszcza dość często stosowaną miarę Lee [2].

Stosując zestaw reguł utworzonych algorytmem GTS do klasyfikacji nowego („nieznanego") przypadku możemy mieć do czynienia z jedną z następujących sytuacji:

- ***tylko jedna reguła*** dotyczy badanego przypadku i jest on poprawnie sklasyfikowany przez tę regułę,
- ***więcej niż jedna reguła*** obejmuje badany przypadek. Wtedy zostaje użyta reguła o największej mierze H, oraz
- ***żadna*** z utworzonych reguł nie może sklasyfikować nieznanego przypadku, lub został on błędnie sklasyfikowany. W tej sytuacji nieznany przypadek należy dodać do zbioru treningowego i restartować proces indukcji reguł.

Działanie algorytmu GTS

Uruchomienie algorytmu GTS jest poprzedzone działaniem automatycznych procedur realizujących wybrane zadania związane z weryfikacją i walidacją danych. Analiza zbioru treningowego może bowiem dostarczyć wiele ważnych informacji, które powinny być sprawdzone przed rozpoczęciem procesu tworzenia modelu klasyfikacji. Najogólniej biorąc, zbierane są m.in. następujące informacje o zbiorze treningowym: liczba wszystkich przypadków, liczba klas, liczba poprawnie opisanych przypadków, liczba przypadków z błędami (wykrywane są błędy pisowni, błędne nazwy, itp.; tego rodzaju przypadki muszą być usunięte ze zbioru treningowego), liczba zgodnych przypadków, liczba przypadków redundantnych, liczba przypadków sprzecznych (również i tu trzeba dokonać wyboru). Proces analizy statusu przypadków w zbiorze treningowym, w następnej fazie dotyczy trudniejszych problemów: poszukiwane są przypadki odległe oraz intencjonalne kłamstwa, które mogłyby zakłócić klasyfikację i ewentualnie doprowadzić do utworzenia fałszywego modelu klasyfikacji. Rozstrzygnięcia w zakresie tych dwóch zagadnień dokonywane są w oparciu o własne rozwiązania w dziedzinie rozpoznawania obrazów (tzw. „look ahead-and-back KNN") i wymagają zastosowania specjalnej procedury wizualizacyjnej, transformującej wielowymiarowe dane (z aż 300 wymiarów) do postaci projekcji 2-wymiarowej. Po zakończeniu analizy danych w zbiorze treningowym, GTS przemieszcza atrybuty numeryczne w skrajne lewe położenie, porządkując ich kolejność wg dynamiki ich wartości. Oznacza to, że atrybut o wyraźnie większych wartościach będzie zajmował skrajne lewe położenie, i od niego rozpocznie się tworzenie modelu klasyfikacji. Etap porządkowania atrybutów ciągłych kończy się obliczeniem wartości progowych używanych w procesie indukowania reguł. Zazwyczaj, punkt środkowy każdego przedziału danych zostaje uznany za wartość progu. Dla atrybutu o **m** wartościach, mamy do zbadania nie więcej od (**m-1**) progów. Może wydawać się zbyt kosztownym próbkowanie wszystkich (**m-1**) wartości progowych w procesie indukcji reguł wnioskowania, jednakże co najmniej trzy ważne argumenty przesądzają o stosowaniu tej procedury: (i) opracowany model klasyfikacji staje się elastyczny i umożliwia wprowadzanie odmiennych wartości argumentów od wartości zapisanych w zbiorze treningowym, (ii) w rzeczywistości, liczba wartości progowych jest mniejsza od teoretycznej liczby (**m-1**), oraz (iii) szybkość komputerów na ogół podwaja się co rok, dlatego dzisiejsze algorytmy, uznane jako czasochłonne, wkrótce będą realizowane prawie w mgnieniu oka.

Porządkując (malejąco) ciągłe atrybuty zbioru treningowego mamy:

2000, 1500, 1500, 1500, 1500, 1200, 1200 (dla atrybutu <Zarobki>)

7, 6, 4, 2, 2, 2, 2 (dla atrybutu <Lata_pracy>).

W pierwszym przypadku mamy zaledwie dwie wartości progowe, a mianowicie 1750 i 1350 (punkt środkowy 1500 pomiędzy dwoma kolejnymi wartościami 1500 nie jest progiem lecz aktualną wartością atrybutu!). W drugim przypadku <Lata_pracy>, mamy jedynie trzy progi: 6.5, 5 oraz 3.

Po wyznaczeniu wartości progowych, rozpoczyna się proces indukowania reguł wnioskowania, w kilku kolejnych krokach:

(*w dalszych rozważaniach, zamiast nazwy atrybutu* <**Wniosek_o_kredyt**>, *jest stosowana krótsza nazwa* <**KREDYT**>)

Krok_1. Wybrany zostaje pierwszy przypadek ze zbioru treningowego. Utworzone zostają następujące wyrażenia logiczne:

		$\sqrt{A}$
1: (Zarobki,<1750	→(KREDYT, zaakceptowany)	0.5774
2:(Zarobki,≥1350	→(KREDYT, zaakceptowany)	0.6325
3:(Lata_pracy,<3)	→(KREDYT, zaakceptowany)	0.7071
4:(Historia_kredytu,doskonała)	→(KREDYT, zaakceptowany)	0.8165
5:(Zabezpieczenie_nieruchomościami)	→(KREDYT,zaakceptowany)	0.8165

Należy zwrócić uwagę, że dla atrybutu <Zarobki> zostały utworzone dwa wyrażenia, ponieważ jego aktualna wartość wynosi 1500 (co jest mniejsze od 1750 i jednocześnie większe od 1350). Atrybut <Lata_pracy> ma jedno tylko wyrażenie, ponieważ jego wartość (2) dla pierwszego przypadku jest jednostronnie ograniczona progiem 3. Algorytm GTS posługuje się wartością najbliższego progu, i z tego względu w rozważanym przypadku pomija inne wartości progowe (5 oraz 6.5).

Krok_2. Ponieważ żadne z wymienionych wyrażeń nie pokrywa wyłącznie jednej klasy, podczas kolejnej iteracji proces indukcji reguł podlega specjalizacji (zgodnie z realizacją koncepcji General-To-Specific). Rozpoczynając od atrybutu ciągłego o największej wartości, utworzone zostaje nowe wyrażenie logiczne:

1:(Zarobki,≥1350) & (Historia_kredytu,doskonała) → (KREDYT, zaakceptowany)

Wybrany został próg ≥1300, gdyż w poprzednim kroku był oceniony lepiej od progu <1750. Bieżące testowanie członu dokładności ujawnia, iż A wynosi 1, i dlatego zostaje wygenerowana reguła:

Reguła 1 (H = 1.2857):

JEŻELI (Zarobki, 1300)
ORAZ (Historia_kredytu,doskonała)
TO (KREDYT,zaakceptowany)

Reguła 1 klasyfikuje przypadek 1 oraz 2. Są one odznaczane jako sklasyfikowane (lecz **nie** usuwane ze zbioru treningowego! Aż do zakończenia procesu tworzenia modelu klasyfikacji, zawartość zbioru treningowego pozostaje niezmieniona).

Krok_3. Następny przypadek (Lp. 3) zostaje wybrany do przetwarzania, i cztery nowe wyrażenia logiczne są rozważane:

1: (Zarobki,<1350) → (KREDYT,odrzucony)
2: (Lata_pracy,<3) → (KREDYT,odrzucony)
3: (Historia_kredytu,doskonała) → (KREDYT,odrzucony)
4: (Zabezpieczenie_nieruchomościami) → (KREDYT,odrzucony)

Ponieważ pierwsze wyrażenie stosuje się tylko do jednej klasy (człon A osiągnął wartość 1), zostaje utworzona nowa reguła:

Reguła 2 (H = 1.1429):

JEŻELI (Zarobki,<1350)
TO (KREDYT,odrzucony)

(*pokryte są przypadki 3 i 5*)

Krok_4, 5, 6, ... Kolejne przypadki ze zbioru treningowego (n i e sklasyfikowane) są oceniane i indukowane kolejne reguły wnioskowania, aż do momentu utworzenia kompletnego modelu klasyfikacji:

Reguła 3 (H = 1.1429):

JEŻELI (Zarobki, 1750)
TO (KREDYT,zaakceptowany)

(*pokryty przypadek 4-ty*)

Reguła 4 (H = 1.1429):

JEŻELI (Zarobki, 1350)
ORAZ (Historia_kredytu,dobra)
ORAZ (Lata_pracy,<5)
TO (KREDYT,odrzucony)

(*pokryty przypadek 6-ty*), i

Reguła 5 (H = 1.1429):

JEŻELI (Historia_kredytu,mierna)
TO (KREDYT,odrzucony)

(*ostatni, 7-my przypadek został sklasyfikowany*).

W tym momencie algorytm kończy działanie, ponieważ wszystkie przypadki ze zbioru treningowego zostały sklasyfikowane. Utworzony model klasyfikacji może być oceniony – w odniesieniu do zbioru treningowego – stosując tą samą co poprzednio miarę, tj. średnią liczbę pytań. Wynosi ona 1.5714; wartość ta jest zbliżona do S_{LERS} i wyraźnie lepsza od S_{ID3}.

Połączenie GTS z MLC

Jakiekolwiek odkrycie dokonane z zastosowaniem określonej metodyki powinno być potwierdzone stosując inną, niezależną i wiarygodną metodologię. Ta obiektywna prawda odgrywa szczególną rolę w naukach technicznych i przyrodniczych, także w biznesie, ekonomii oraz zarządzaniu. Dokonanie błędnych spostrzeżeń lub odkryć i implementacja fałszywych wniosków, prowadzą – we wszystkich dziedzinach nauki i techniki – do katastrofy. Dotyczy to także, a może nawet szczególnie modeli klasyfikacyjnych utworzonych z zamysłem odkrywania regularności w danych. Z tego też względu wydało się celowym spotęgowanie skuteczności działania algorytmu GTS poprzez złączenie z innymi metodami uczenia maszynowego. Zwróciliśmy uwagę na nieco zapomnianą, odzyskującą obecnie coraz większe znaczenie metodę KNN (K-th Nearest Neighbour), znaną pod tą nazwą statystykom, a stosowaną także przez naukowców pracujących w dziedzinie sztucznej inteligencji. Stanowi ona podstawę metod uczenia maszynowego, znanych w informatyce m.in. jako „rozumowanie oparte na precedensach" (ang.: case-based reasoning). Metoda KNN (w wersji „look-ahead-and-back", opracowanej w naszej grupie) w połączeniu z algorytmem GTS, utworzyła interesujące teoretycznie i obiecujące praktycznie możliwości elastycznej weryfikacji modelu klasyfikacji, stosowanego do poszukiwania regularności w danych. Jednym z problemów, który wymagał rozwiązania przed dokonaniem implementacji tego zamysłu, była konwersja wartości atrybutów (symboliczne numeryczne), wspomniana już w rozdziale 3.2, gdyż algorytm MLC akceptuje wyłącznie dane numeryczne. Najprostszym sposobem zrealizowania tej konwersji jest mapowanie wartości dyskretnych liczbami całkowitymi (np. przypisując 3 = doskonała, 2 = dobra, 1 = mierna atrybutowi <Historia_kredytu>). W ten sposób zbiór treningowy, dotyczący problemu aplikacji o kredyt bankowy będzie zapisany w następującym formacie:

Lp.	Historia_kredytu	Lata_pracy	Zarobki [PLN]	Zabezpieczenie_ nieruchomościami	Wniosek_o_KREDYT
1	3	2	1.5	2	zaakceptowany
2	3	2	1.5	1	zaakceptowany
3	3	2	1.2	2	odrzucony
4	2	6	2.0	2	zaakceptowany
5	2	7	1.2	1	odrzucony
6	2	4	1.5	1	odrzucony
7	1	2	1.5	1	odrzucony

(Wartości atrybutu <Zarobki> zostały przekształcone do formatu [wartość] x 1000, w celu zrównania dynamiki danych)

Szczegóły nt. algorytmu MLC są podane np. w [1]. Tutaj, należy jedynie wyjaśnić, że algorytm MLC ujmuje dwie różne metody poszukiwania regularności w danych. Pierwsza z nich, w zastosowaniu do zbioru treningowego, wykorzystuje ogólną strategię:

[warunki (arbitralne)] → <rekomendacje: KREDYT,zaakceptowany
KREDYT,odrzucony>

Druga strategia, nazwana właśnie „look ahead-and-back" umożliwia wędrowanie po reprezentacji danych (przypadków) z *lewa-w-prawo:*

[warunki (arbitralne)] → <rekomendacje: KREDYT, zaakceptowany
KREDYT, odrzucony>

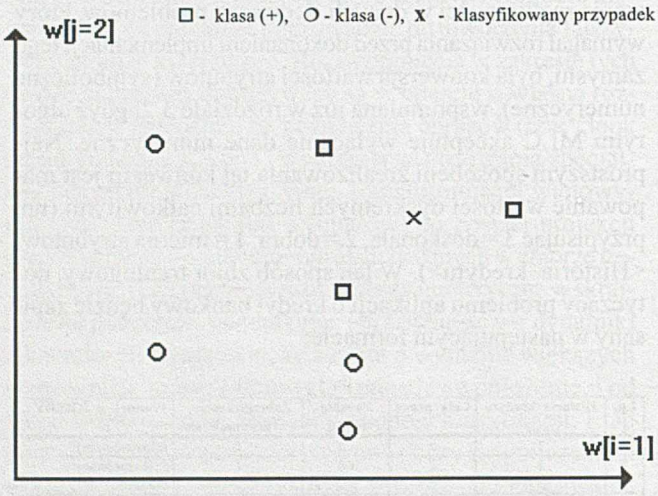
oraz z *prawa-na-lewo:*

<zalecone (rekomendowane) warunki>← [charakterystyka wniosku o udzielenie kredytu].

Pierwsza strategia, po wstępnym przetwarzaniu danych w zbiorze treningowym za pomocą narzędzi dostępnych w ramach algorytmu MLC (wszystkie dane zostały przeliczone stosując funkcję

$$f(x) = \sqrt{A}$$

a następnie auto-skalowanie) dostarczyła bardzo interesujące wyniki. Zastosowanie procedury wizualizacyjnej, spłaszczającej czterowymiarowe dane (cztery atrybuty zastosowane w zbiorze treningowym) do płaszczyzny 2D, lepiej rozdzieliło charakterystyczne obszary podobieństwa. Utworzone zostały dwa wyraźne skupiska, grupujące przypadki pozytywne (KREDYT,zaakceptowany) lub negatywne (KREDYT,odrzucony). Co więcej, można teraz wprowadzić dowolny przypadek i przypisać mu najbardziej prawdopodobną decyzję. Nie jest to możliwe w wypadku stosowania modelu sformalizowanego przy pomocy reguł wnioskowania.



Rys. 1. Reprezentacja 2D zbioru treningowego.

W[i] oraz w[j] są wektorami własnymi macierzy kowariancji, □, ○ – są punktami odpowiadającymi przypadkom w zbiorze treningowym, x – jest nieznanym przypadkiem (z wartościami 2.5, 3, 1.7 oraz 1.7 kolejnych atrybutów). Widać, że rekomendacją dla rozpatrywanego przypadku będzie <KREDYT,zaakceptowany>. Gdy punkt reprezentujący nieznany przypadek jest oddalony od głównych obszarów skupisk reprezentujących zbiór treningowy, użytkownik podejmuje decyzję czy aktualny zbiór treningowy może być stosowany do klasyfikacji nieznanego przypadku.

Druga strategia, w wersji z „lewa-w-prawo" (do przodu) pozwala wyciągnąć te same wnioski, które otrzymano poprzednio. Jednakże, możliwą jest tu także klasyfikacja wielokategoryjna, powiedzmy<KREDYT,odrzucony>,<KREDYT,zaakceptowany>, <KREDYT,zaakceptowany_po_N_miesiącach>, itp. Natomiast rozumowanie w przeciwnym kierunku (wstecz, z „prawa-na-lewo") pozwala uzyskać kompetentne odpowiedzi (stopień kompetencji zależy od jakości i rozmiarów zbioru treningowego) na pytania rodzaju:

1) *jakie są ogólne warunki akceptacji wniosku o kredyt, z zachowaniem właściwego marginesu bezpieczeństwa banku?*

oraz

2) *w jakich warunkach bank (anonimowy) jest zmuszony do odrzucenia wniosku o kredyt?*

Kontynuując eksperymenty ze zbiorem treningowym użytym w niniejszym artykule, wykryto następujące zależności:

	Pytanie 1 (pozytywne)	Pytanie 2 (negatywne)
Historia_kredyt	2.4	2.1
Lata_pracy	3.5	3.7
Zarobki	1600.-	1400.-
Zabezpieczenie_nieruchomościami	1.5	1.3

Dane te, po ponownej konwersji atrybutów (uprzednio dyskretnych), zawierają następujące rekomendacje:

Historia_kredyt	lepsza od dobrej (80%)	dobra
Lata_pracy	ponad 3 lata	mniej od 4 lat
Zarobki	powyżej 1600.- PLN	mniej od 1350.- PLN
Zabezpieczenie_nieruchomościami	powyżej 75%	mniej od ~ 65%

Podsumowując należy podkreślić dobrą zgodność modelu klasyfikacji utworzonego wyłącznie przy pomocy algorytmu GTS, z modelem uzupełnionym wiarygodną procedurą algebraiczną.

Wymagane uzupełnienia

Zawsze ulegamy pokusie dodania kolejnych cech lub/oraz specjalnych funkcji do opracowanego systemu informatycznego. Podobnie jest z narzędziem programowym realizującym algorytm GTS. Wydaje się, iż korzystnym usprawnieniem rozszerzającym możliwości tworzenia modeli klasyfikacyjnych byłaby implementacja mechanizmów umożliwiających przetwarzanie atrybutów, które mogą mieć wartości nieznane, w rodzaju: <nie_wiem>, <każdy>, <dowolny>, <wszystko_jedno>, itp. Inną pożyteczną funkcją byłaby zdolność przetwarzania atrybutów strukturalnych (tj. atrybutów mających dopuszczalne wartości raczej w formie zhierarchizowanej, niż w postaci listy).

Dyskusja

Można z uzasadnionym optymizmem oczekiwać lepszych skutków poszukiwania regularności w bazach danych przy pomocy algorytmów pokrycia, wzmocnionych podejściem opartym na modelach algebraicznych. Aczkolwiek wielu naukowców zajmujących się uczeniem maszynowym może odnieść się do tej koncepcji ze zrozumiałym krytycyzmem, mamy przekonanie, że istnieje liczna grupa praktyków sceptycznie zapatrujących się na regułowy model klasyfikacji danych. Należy jednak koniecznie zaakcentować jedną z najważniejszych pozytywnych cech algebraicznego wzmocnienia budowy modeli klasyfikacyjnych: stosując to rozwiązanie, można zapomnieć o wszystkich frustrujących problemach generowania całkowicie odmiennych drzew decyzji (lub zestawów reguł wnioskowania), jeśli zostanie zmieniona kolejność atrybutów w zbiorze treningowym!

Podsumowując, należy wyrazić przekonanie, że ostrożne i rozsądne zastosowanie algorytmu GTS wzmocnionego algebraicznym modelem klasyfikacji, może stanowić użyteczną metodologię w arsenale narzędzi badawczych, stosowanych do przetwarzania danych w biznesie, marketingu i zarządzaniu.

Podziękowanie

Chciałbym wyrazić wdzięczność wielu osobom zaangażowanym w realizacji opisanego projektu badawczego. Szczególnie dotyczy to: mgr M. Mazura, za doprowadzenie z wielką cierpliwością modułu MLC/SCANKEE do aktualnej postaci, dr hab. B. Dębskiej za wkład w opracowanie metody wizualizacji wielowymiarowych danych, oraz T. M. Hippe, który dopracował szczegóły algorytmu GTS oraz zakodował i uruchomił jego wersję programową.

Przedstawione wyniki są częścią projektu badawczego (Nr 8 T11C 004 09) finansowanego przez Komitet Badań Naukowych. Bez tego ważnego dla nas wsparcia nie byłoby możliwym zrobienie opisanego małego postępu w wybranych dziedzinach sztucznej inteligencji.

Literatura

[1] Hippe, Z.S. (1996). ***Multistrategy reasoning hybrid expert system of open architecture***. Proc. 3rd World Congress on Expert Systems, Seoul 4-9.02., pp. 1171-1178

[2] Lee, C. (1994). ***Generating classification rules from databases***. Proc. 9th Conf. on Appl. of AI in Engineering, Malvern (PA), pp. 205-212.

prof. Zdzisław Hippe jest pracownikiem Katedry Informatyki Chemicznej, Politechniki Rzeszowskiej.

Prawdziwe informacje kosztują...

Biuro DiS prezentuje najobszerniejsze w swoich tematach komercyjnie oferowane raporty:

- OWZP' 97, raport oprogramowanie **Wspomagające Zarządzanie Przedsiębiorstwami**: zestawienie wymagań, opisy 340 produktów dla firm każdego rodzaju (280 stron)
- KZMRP' 97 **Krajowe i Zagraniczne Systemy MRP**: software dla przedsiębiorstw produkcyjnych i dystrybucyjnych, badania i analizy wdrożeń (208 stron plus plansza z tubą)
- ROU' 97 **Rynek Oprogramowania i Usług**, pierwsza kompleksowa próba oceny jednej z najciekawszych i najdynamiczniej się rozwijających nisz rynkowych informatyki (wyd. maj 1997, przewidywana objętość 250 stron)
- GSI' 97, **Globalne Społeczeństwo Informacyjne**, aspekty wdrożenia w kraju, wnioski i analizy, raport niesponsorowany (wyd. lipiec 1997, przewidywana objętość 100 stron)

DiS, tel/fax (0-22) 7799069, Internet: dis1@pol.pl

...pozostałe są rozdawane

SYSTEMY CZASU RZECZYWISTEGO

REDAGUJE ZESPÓŁ: dr inż. Zbigniew FRYŹLEWICZ (Politechnika Wrocławska), prof. dr hab. inż. Janusz GÓRSKI (Instytut Technik Telekomunikacyjnych i Informatycznych w Poznaniu), prof. dr hab. inż. Zbigniew HUZAR (Politechnika Wrocławska), prof. dr hab. inż. Tomasz SZMUC (Akademia Górniczo-Hutnicza), dr inż. Kazimierz WORWA (Wojskowa Akademia Techniczna), mgr inż. Zdzisław ŻURAKOWSKI (Instytut Automatyki Systemów Energetycznych we Wrocławiu) - przewodniczący zespołu.

ROK IV • MAJ • 1997 • 28

Języki programowania

Programowanie systemów czasu rzeczywistego w Adzie 95

Część II

Zbigniew Fryźlewicz, Zbigniew Huzar

Priorytety dynamiczne

Często występują sytuacje, w których założenie stałości priorytetów nie jest właściwe. Przykładem mogą być:

- potrzeba zmiany priorytetu zadania na skutek zmiany jego trybu pracy;
- szeregowanie zadań wg. strategii EDF.

W takich sytuacjach niezbędne stają się zmiany priorytetów bazowych zadań. W celu przeprowadzania takich zmian zdefiniowano pakiet biblioteczny Ada.Dynamic_Priorities. Ma on następującą specyfikację:

```
with Ada.Task_Identification;
with System;
package Ada.Dynamic_Priorities is
procedure Set_Priority (Priority : System.Any_Priority;
                        T : Ada.Task_Identification.Task_Id :=
                Ada.Task_Identification.Current_Task);
          — generuje wyjątek Tasking_Error, gdy wskazano
               nieistniejące lub zakończone zadanie

function Get_Priority (T : Ada.Task_Identification.Task_Id :=
                       Ada.Task_Identification.Current_Task)
                              return System.Any_Priority;
          — generuje wyjątek Tasking_Error, gdy wskazano
               nieistniejące lub zakończone zadanie

private
      — określone przez implementację
end Ada.Dynamic_Priorities;
```

Do wskazywania określonych zadań, pakiet wykorzystuje ich identyfikatory. Funkcja Get_Priority zwraca aktualną wartość bazowego priorytetu wskazanego zadania. Procedura Set_Priority ustala nowy priorytet bazowy wskazanego zadania. Realizacja zmiany priorytetu bazowego zależy od implementacji i następuje tak szybko jak jest to możliwe w danym środowisku wykonawczym. Brak precyzyjnego określenia wynika z tego, że priorytety zadań, w danej implementacji, mogą być uzależnione od lokalnego systemu operacyjnego. Konkretna implementacja języka precyzuje momenty zmian priorytetów zadania.

Zmiana priorytetu bazowego wpływa na zmiany priorytetu aktywnego zadania i tym samym ma wpływ na szeregowanie zadań. W obrębie predefiniowanych metod szeregowania zmiana priorytetu zadania powoduje jego przeniesienie na koniec kolejki związanej z odpowiednim priorytetem.

Mechanizm priorytetów dynamicznych umożliwia realizację własnych strategii szeregowania zadań. Jako przykład ilustrujący te możliwości rozpatrzmy szeregowanie zbioru wykonujących się cyklicznie zadań. Każde z zadań co pewien odcinek czasu równy Okres musi wykonać pewne działanie. W danym momencie czasu zadanie, które ma najwcześniejszy termin wykonania pewnego działania, staje się zadaniem o najwyższej pilności i – zgodnie z metodą EDF - zostaje wybrane do realizacji. Przedstawiony poniżej schemat programu zakłada, że szeregowane zadania sterują się same, wykorzystując obiekt chroniony Dyspozytor. Dyspozytor za pomocą procedury Przelicz zmienia dynamicznie priorytety zadań.

```
with Text_IO; use Text_IO;
with Ada.Task_Identification;

with  Ada.Task_Attributes;
— Ada.Task_Attributes jest pakietem generycznym, który po-
zwala na przyporządkowanie
```

```
— zadaniom wybranego atrybutu (typ atrybutu jest parametrem
generycznym pakietu)
— oraz dostarcza operacje nadawania wartości i testowania
wartości atrybutu.

with Ada.Real_Time; use Ada.Real_Time;
— Pakiet Ada.Real_Time jest pakietem definiującym czas
w postaci odpowiedniej
— do zastosowań w czasie rzeczywistym.

with Ada.Dynamic_Priorities, System;
use Ada;

procedure EDF is
— Procedura EDF jest procedurą główną programu, zawiera
szeregowane zadania - typu Wykonaw-
— ca oraz obiekt Dyspozytor wykonujący podstawowe funkcje
związane z szeregowaniem

pragma Locking_Policy (Ceiling_Locking);
pragma Task_Dispatching_Policy (Fifo_Within_Priorities);

N : constant Positive := 3;  — N jest maksymalną liczbą
szeregowanych zadań
subtype Aktywne_zadania is Natural range 0 .. N;
subtype Indeksy_zadan is Natural range 1 .. N;
Wysoki : constant System.Priority := System.Priority'Last;
Sredni : constant System.Priority := Wysoki - 1;
Niski : constant System.Priority := Wysoki - 2;

package Atrybuty_przeterminowania is new Task_Attributes(Time,
Time_Last);
— Pakiet Atrybuty_przeterminowania służy do przypisania
zadaniom atrybutu czasowego,
— którym są odpowiadające im chwile przeterminowania. Time
jest parametrem aktualnym, który
— jest typem prywatnym zdefiniowanym w pakiecie
Ada.Real_Time.
— Time_Last jest stałą określoną przez implementację w pakiecie
Ada.Real_Time.
— Stała ta określa najpóźniejszy moment dający wyrazić się
w danej implementacji

type Etykiety is array(Indeksy_zadan) of Task_identification.Task_Id;

protected Dyspozytor is
pragma Priority(Wysoki);
— Obiekt chroniony Dyspozytor ma przypisany najwyższy
priorytet w danej implementacji

procedure Rejestruj;
— Wywołanie procedury przez zadanie powoduje dopisanie
indentyfikatora
— tego zadania do zbioru szeregowanych zadań

procedure Przelicz;
— Procedura jest wywoływana przez zadania w momencie
opuszczania przez nie kolejki zadań
— oczekujących na dopuszczenie do realizacji

procedure Usun;
— Procedura jest wywoływana przez szeregowane zadanie po
wykonaniu przez nie wszystkich
— cyklicznych działań

private
   Etykiety_zadan : Etykiety;
   Liczba_zadan : Aktywne_zadania := 0;
   Najkrotszy_termin : Time := Time_Last;
   Zadanie_aktywne : Task_Identification.Task_Id := Task_Identifi-
   cation.Null_Task_Id;
end Dyspozytor;
```

```
task type Wykonawca is
   pragma Priority(Wysoki);
   entry Start(Cykl: in Time_Span; Dlugosc: in Time_Span);
end Wykonawca;
   — Typ ten jest nadbudową, w której są zanurzone treści
   szeregowanych zadań.

protected body Dyspozytor is
procedure Rejestruj is
begin
   Liczba_zadan := Liczba_zadan + 1;
   Etykiety_zadan(Liczba_zadan) := Task_Identification.Current_Task;
   Zadanie_aktywne := Task_Identification.Current_Task;
end Rejestruj;

procedure Przelicz is
begin
   if  Atrybuty_przeterminowania.Value < Najkrotszy_termin
   then
      Dynamic_Priorities.Set_Priority(Sredni);
      Dynamic_Priorities.Set_Priority(Niski,Zadanie_aktywne);
      Najkrotszy_termin := Atrybuty_przeterminowania.Value;
      Zadanie_aktywne := Task_Identification.Current_Task;
   else
      Dynamic_Priorities.Set_Priority(Niski);
   end if;
end Przelicz;

procedure Usun is
begin
   Najkrotszy_termin := Time_Last;
   Zadanie_aktywne := Task_Identification.Null_Task_Id;
   for T in 1 .. Liczba_zadan loop
      if Atrybuty_przeterminowania.Value(Etykiety_zadan(T))
      < Najkrotszy_termin
      then
            Najkrotszy_termin := Atrybuty_przeterminowania.
            Value(Etykiety_zadan(T));
            Zadanie_aktywne := Etykiety_zadan(T);
      end if;
   end loop;
   if  Najkrotszy_termin < Time_Last
   then
      Dynamic_Priorities.Set_Priority(Sredni, Zadanie_aktywne);
   end if;
end Usun;
end Dyspozytor;

task body Wykonawca is
   Okres: Time_Span;
   Termin : Time_Span :=To_Time_Span(0.010);  —przykładowa
   wartość przeterminowania
   Nastepny_cykl : Time;
   Koniec : Time;
begin
   accept Start(Cykl: in Time_Span; Dlugosc: in Time_Span)
   do  —czekaj na start
      Okres := Cykl;
      Koniec := Clock + Dlugosc;
   end Start;
   Dyspozytor.Rejestruj;
   Nastepny_cykl := Clock + Okres;
   Dynamic_Priorities.Set_Priority(Wysoki);
   loop
      delay until (Nastepny_cykl);
      Atrybuty_przeterminowania.Set_Value(Nastepny_cykl +
      Termin);
      Dyspozytor.Przelicz;
```

```
      ...                                  — Treść szeregowanego zadania.
      exit when Clock > Koniec; — zakończ obliczenia cykliczne
                                  gdy czas minął
      Dynamic_Priorities.Set_Priority(Wysoki);
      Atrybuty_przeterminowania.Set_Value(Time_Last);
      Nastepny_cykl := Nastepny_cykl + Okres;
   end loop;
   Dyspozytor.Usun;
end Wykonawca;

w1, w2, w3: Wykonawca;   — Deklaracje zadań typu Wykonawca
begin
   w1.Start(To_Time_Span(3.00), To_Time_Span(20.0));
   — start kolejnych zadań z przy-
   w2.Start(To_Time_Span(3.25), To_Time_Span(21.0));
   — kładowymi okresami i czasem
   w3.Start(To_Time_Span(3.50), To_Time_Span(22.0));
   — trwania cyklicznych obliczeń
end EDF;
```

Przedstawione rozwiązanie ma charakter ilustracyjny. Można je usprawnić np. przez budowę kolejki zadań, zgodnie z upływającymi czasami przeterminowań. Spowoduje to przyspieszenie działania procedury Usun, ale wymaga także zmian w procedurze Przelicz. Ulepszone rozwiązanie jest przedstawione w [4].

Synchronizacja i sterowanie zadaniami

Ada 83 zawiera mechanizmy synchronizacji tylko wysokiego poziomu. Utrudnia to czasem tworzenie systemów silnie uwarunkowanych czasowo. W Adzie 95 wprowadzono dwa dodatkowe pakiety, zdefiniowane w aneksie RTS, zawierające mechanizmy synchronicznego i asynchronicznego sterowania zadaniami.

Synchroniczne sterowanie zadaniami

Predefiniowany pakiet synchronicznego sterowania zadaniami udostępnia mechanizm przypominający semafor binarny. Specyfikacja pakietu jest następująca:

```
package Ada.Synchronuos_Task_Control is
   type Suspension_Object is limited private;
   procedure Set_True(S : in out Suspension_Object);
   procedure Set_False(S : in out Suspension_Object);
   function Current_State Set_True(S : Suspension_Object)
   return Boolean;
   procedure Suspended_Until_True(S : in out
   Suspension_Object);
   — procedura generuje wyjątek Program_Error jeżeli więcej
   niż jedno zadanie próbuje
   — jednocześnie zawiesić swe działanie ze względu na S
private
   — część nie definiowana w języku
end Ada.Synchronuos_Task_Control;
```

Pakiet oferuje operacje na zmiennych typu Suspension_Object. Zmienne tego typu mają dwa stany prawda i fałsz, ustawiane procedurami Set_True i Set_ False. Opracowanie deklaracji zmiennej tego typu powoduje przyjęcie przez nią wartości fałsz. Funkcja Current_State_Set udostępnia aktualną wartość zmiennej. Procedury Set_True i Set_False nie powodują zawieszania wywołujących je zadań. Procedura Suspend_Until_True powoduje zawieszanie zadania, które ją wywołuje, aż do momentu gdy odpowiednia zmienna typu Suspension_ Object przyjmie wartości prawda. W tym momencie stan zadania ulega zmianie z „zawieszone" na „gotowe" z równoczesną zmianą wartości zmiennej z prawda na fałsz.

Zmienną typu Suspention_Object można traktować jako prywatny semafor binarny zadania. Prywatność oznacza tu, że w dowolnym wycinku czasu, ze względu na wartość zmiennej tego typu, może być zawieszone tylko pojedyncze zadanie. Jeżeli warunek ten nie jest spełniony, to generowany jest wyjątek Program_Error. Realizacja tak określonego semafora jest efektywna czasowo, gdyż nie wymaga utrzymywania kolejki zablokowanych zadań.

Dysponując opisanym typem i operacjami można łatwo zaprogramować wykluczający się dostęp dwóch zadań do wspólnych danych dzielonych. Inne przykłady zastosowań są opisane w [5] i [6].

Asynchroniczne sterowanie zadaniami

Pakiet asynchronicznego sterowania zadaniami udostępnia mechanizm zawieszania innych zadań. Zadanie może zawiesić inne zadanie przez obniżenie jego bazowego priorytetu do tzw. priorytetu zawieszenia. Pakiet ma następującą specyfikację:

```
with Ada.Task_Identification;
package Ada.Asynchronous_Task_Control is
   procedure Hold (T : Ada.Task_Identification.Task_Id);
   procedure Continue (T : Ada.Task_Identification.Task_Id);
   function Is_Held(T : Ada.Task_Identification.Task_Id) return
   Boolean;
end T : Ada.Asynchronous_Task_Control;
```

Procedura Hold powoduje obniżenie bazowego priorytetu zadania poniżej poziomu wszystkich zadań w programie, a więc poniżej wartości System.Any_Priority'First. Zadanie otrzymuje priorytet zawieszenia. Procedura Continue przywraca poprzedni priorytet, a funkcja Is_Held informuje o tym, czy zadanie jest zawieszone.

Zawieszenia zadania przez obniżenie priorytetu bazowego umożliwia przejrzyste definiowanie semantyki. Oznacza to, że:

- Jeżeli zadanie zawieszane jest w trakcie wykonywania się w obrębie obiektu chronionego, to jego zawieszenie jest *odłożone do momentu opuszczenia obiektu chronionego.*
- Jeżeli zawieszane zadanie jest w fazie spotkania z innym zadaniem, to jego zawieszenie jest *odłożone do czasu zakończenia spotkania.*
- Jeżeli zawieszane zadanie znajduje się w fazie oczekiwania na obsługę wejścia, a w momencie wywołania było to wejście otwarte, to zostanie wykonana treść odpowiedniej instrukcji **accept**. Po zakończeniu spotkania nastąpi obniżenie aktywnego priorytetu zadania-serwe-

ra do poziomu priorytetu wstrzymania (pod warunkiem, że nie dziedziczy ono priorytetu z innych źródeł) i zadanie nie wykonuje się do czasu wykonania na nim operacji Continue.

- Podobne działania mają miejsce, gdy zawieszane zadanie jest jedynym zadaniem w kolejce obiektu chronionego na wejściu z otwartą barierą. Nastąpi wykonanie instrukcji związanych z odpowiednim wejściem.

Zrywanie zadań

Nieodwracalną formą zawieszenia zadania jest jego zerwanie, realizowane przez instrukcję **abort**. Zerwanie powinno być realizowane natychmiast lub, co najmniej, w dającym się oszacować czasie. W Adzie 95 zrywanie musi być niekiedy odłożone w czasie, ze względu na konieczność zachowania integralności całego programu. Zrywanie zadania jest odkładane, gdy zadanie znajduje się w fazie wykonywania w obrębie obiektu chronionego lub w fazie spotkania z innym zadaniem. Ogólnie, zerwanie zadania jest semantycznie zgodne z wykonaniem na nim operacji Hold.

Uproszczenia modelu zadaniowego

Jeżeli wykonanie programu ma być efektywne, a program nie korzysta ze wszystkich mechanizmów języka, to ze środowiska wykonawczego można usunąć niepotrzebne elementy. Aneks RTS zawiera zestaw 13 pragm służących do wyrażania ograniczeń i uproszczeń modelu zadaniowego przyjętego w budowanym programie. Przykładowo:

- No_Task_Hierarchy oznacza, że wszystkie zadania zależą bezpośrednio od zadania środowiska wykonawczego,
- No_Abort_Statement oznacza, że w programie nie występuje instrukcja zrywania zadań **abort**,
- No_Dynamic_Priorities oznacza, że priorytety zadań programu nie są dynamicznie zmieniane,
- No_Asynchronous_Control oznacza, że program nie korzysta z opcji asynchronicznego sterowania zadaniami,
- Max_Select_Alternatives określa maksymalną liczbę gałęzi w instrukcji **select**,
- Max_Task określa maksymalną liczbę zadań w programie,
- No_Implicit_Heap_Allocation brak jest operacji, które niejawnie żądają przydziału pamięci w formie sterty.

Wprowadzenie uproszczeń i ograniczeń modelu zadaniowego wymaga prowadzenia kontroli czy nie ma przypadków ich naruszenia. Dla wielu ograniczeń jest ona przeprowadzana podczas kompilacji programu, część ograniczeń wymaga kontroli podczas wykonywania się programu. Program z uproszczonym modelem zadaniowym ma mniejszy zakres niedeterminizmu, co ułatwia jego analizę.

Zegary i czas

Wszystkie wbudowane systemy SCR muszą koordynować swoją aktywność z upływem naturalnego czasu. W tym celu muszą mieć dostęp do układowego zegara, który z pewną dokładnością i rozdzielczością przybliża upływający czas naturalny. W systemach o długim cyklu życia zegar układowy musi być co pewien czas koordynowany z zewnętrznym źródłem pomiaru czasu. Ada udostępnia dwa pakiety z funkcjami zegarowymi. Pakiet pierwszy, istniejący już w Adzie 83, to Ada. Calendar. Dostarczany przez niego czas ma dwie wady: niezbyt dużą gwarantowaną rozdzielczość i możliwość zmiany wskazywanego czasu „w przód i w tył". Pozbawiony tych wad jest zegar udostępniany przez opcjonalny pakiet Ada.Real_Time. Specyfikacja tego pakietu zawiera, m.in., następujące elementy.

```
package Ada.Real_Time is
type Time is private;
Time_First: constant Time;
Time_Last: constant Time;
Time_Unit: constant := liczba rzeczywista zależna od
implementacji;

type Time_Span is private;
Time_Span_First: constant Time_Span;
Time_Span_Last: constant Time_Span;
Time_Span_Zero: constant Time_Span;
Time_Span_Unit: constant Time_Span;

Tick: constant Time_Span;
function Clock return Time;
  ...
function Nanoseconds (Ns: Integer) retrun Time_Span;
function Microseconds (Us: Integer) return Time_Span;
function Milliseconds (Ms: Integer) return Time_Span;

type Seconds_Count is range zdefiniowane w implementacji
procedure Split(T: in Time; Sc out Seconds_Count; Ts: out
Time_Span);
function Time_of(Sc: Seconds_Count; Ts_Time_Span) return
Time;

private
— poza definicja języka
end Ada.Real_Time;
```

Typ Real_Time.Time jest przybliżeniem czasu rzeczywistego, zwracanym przez funkcję Real_Time.Clock. Stała Time_Unit jest najkrótszym odcinkiem czasu reprezentowanym przez typ Time z gwarancją, że w każdej implementacji jest to nie więcej niż 20 mikrosekund. Do odmierzania odcinków czasu służy typ Time_Span. Istnieje wzajemnie jednoznaczne odwzorowanie między wartościami tego typu a implementowanym zbiorem liczb całkowitych. Wartość typu Time_Span odpowiadająca liczbie I reprezentuje odcinek czasu rzeczywistego o długości I*Time_Unit. Minimalnym zakresem typu Time jest co najmniej 50 lat (od momentu startu programu). Poza opisanymi typami pakiet Ada.Real_Time dostarcza innych operacji na wartościach typu Time i Time_Span.

❖ ❖ ❖

Z punktu widzenia wytwarzania systemów czasu rzeczywistego, modernizacja Ady 83, która doprowadziła do powstania Ady 95 objęła, następujące elementy.

Pierwszy - dotyczy modelu zadań, priorytetów i czasu. W Adzie 83, ze względu na regułę FIFO przy obsłudze kolejek wejściowych i brak dynamicznych zmian priorytetów, nieuniknione było występowanie inwersji priorytetów. Za-

dania cykliczne można było programować jedynie przy użyciu instrukcji **delay** *czas_względny*, co powodowało narastanie dryftu czasowego. W język była wbudowana tylko jedna strategia szeregowania zadań, nie dając wsparcia do budowy aplikacji z własnymi strategiami szeregowania.

Drugi – dotyczy modelu obsługi przerwań. W Adzie 83 przyjęto, że sterownik przerwania jest zadaniem i pojawienie się przerwania wywołuje wejście tego zadania. Dodatkowo, za pomocą klauzul reprezentacji, realizowano dostęp do rejestrów urządzenia. Model ten był krytykowany z powodu skomplikowania i wynikających stąd nieefektywnych (czasowo) implementacji. W Adzie 95 przerwania są obsługiwane przez procedury obiektu chronionego. Zlikwidowano również ograniczenia związane z klauzulami reprezentacji.

Trzeci element wiąże się z systemami asynchronicznymi. Interakcja pomiędzy zadaniami jest oparta o mechanizm asymetrycznych, synchronicznych spotkań. Interakcje asynchroniczne należało tworzyć w oparciu o dodatkowe zadania pośredniczące. Na skutek powiększenia liczby zadań efektywność implementacji budowanej aplikacji ulegała pogorszeniu. Dodatkowo, nie było mechanizmów asynchronicznego sterowania wykonaniem zadań (zadanie można było tylko zerwać lub przez dodatkowo wprowadzone wejście sprawdzać żądania „asynchronicznej" komunikacji). W Adzie 95 wprowadzono nowy model asynchronicznej komunikacji międzyzadaniowej w oparciu o statyczne typy chronione [8] oraz w oparciu o asynchroniczną postać instrukcji **select** [10].

Zagadnienie ostatnie to efektywność budowanych aplikacji, wynikająca z przyjętego modelu zadaniowego. Konieczność zachowania zgodności z Adą 83 wykluczyła uproszczenie modelu zadaniowego. Wprowadzono jednak wiele środków ułatwiających budowę efektywnych czasowo aplikacji. Są to:

- komunikacja i synchronizacja za pomocą obiektów chronionych,
- pakiety i udogodnienia dostarczane przez aneks RTS,
- uproszczenia systemów wykonawczych wymuszane pragmami.

Ada 95 jest standardem ISO, co nadaje językowi wysoką rangę. Szerokie rozpowszechnienie Ady, poza stałym wsparciem Departamentu Obrony USA, zależy obecnie od dwóch czynników:

- od powodzenia aktualnych wysiłków prowadzących do produkcji publicznie dostępnych kompilatorów Ady [1, 4]
- od rozwoju zintegrowanych środowisk programistycznych (narzędzi typu CASE).

Literatura

[1] *Ada Information Clearinghouse*, http://sw-eng.falls-church.va.ua/AdaIC/compilers/

[2] *Ada Language Reference Manual*. ANSI/ISO/IEC-8652:1995. February 1995

[3] *Ada 95 Rationale*. Intermetrics, Inc., 1995

[4] *Ada 95 serwer*, http://ci.pwr.wroc.pl./wzi/ada95.html

[5] Barnes J., *Programming in Ada 95*, Addison-Wesley Publ. Comp. 1995

[6] Burns A., Wellings A., *Ada95: An Effective Concurrent Programming Language*, Reliable Software Technologies - Ada-Europe '96, Springer-Verlag 1996

[7] Complak W., Nawrocki J.R., *Real-Time Euclid: Język programowania systemów silnie uwarunkowanych czasowo*, Informatyka, 10/1996

[8] Fryźlewicz Z., *Typy z protekcją w języku Ada 95*, Informatyka 11/1995

[9] Halang W.A., Sacha K.M., *Real-Time Systems*, World Scientific, 1992

[10] Hnatkowska B., Huzar Z., *Implementacja komunikacji czasu rzeczywistego w Adzie 95*, Materiały Konferencyjne Systemy Czasu Rzeczywistego'96, s. 138-149, Oficyna Wydawnicza Politechniki Wrocławskiej, 1996.

dr Zbigniew Fryźlewicz i prof. Zbigniew Huzar
są pracownikami Wydziału Informatyki i Zarządzania, Politechniki Wrocławskiej

Dział Systemy Czasu Rzeczywistego jest dofinansowany przez Komitet Badań Naukowych.

Języki programowania czasu rzeczywistego

PODSUMOWANIE

Zbigniew Huzar

Niniejszy artykuł zamyka cykl poświęcony językom programowania czasu rzeczywistego. Przedstawiono pięć wybranych języków programowania: Real-Time Euclid [3], Occam [10], Chill [13], Pearl [12] oraz Ada-95 [4]. Nasuwają się pytania: w jakim stopniu wybór ten jest reprezentatywny oraz jak wypada porównanie tych języków pomiędzy sobą.

Historia rozwoju języków programowania czasu rzeczywistego została zarysowana w [7], zaś szczegółowo jest przedstawiona, m.in. w [4, 10]. Pierwsze języki specjalnie projektowane do programowania systemów czasu rzeczywistego zaczęły powstawać jeszcze na początku lat sześćdziesiątych. Liczbę języków, które powstały od tego okresu, szacuje się na kilkaset, a może nawet ponad tysiąc. Przegląd najważniejszych języków, które powstały do końca lat siedemdziesiątych za-

wiera książka Glassa [5], a jednym z ostatnich obszernych przeglądów, obejmującym okres do początku lat dziewięćdziesiątych, jest artykuł Stoyenki [11]. Analizę porównawczą sześciu wybranych języków (Ada 83, FORTRAN, LTR, PL/1, RT Euclid) przedstawiono w książce Halanga i Sachy [6].

Dokonany wybór języków można uzasadnić ich pozytywnymi właściwościami oraz ich rozpowszechnieniem w sensie praktycznego stosowania lub traktowania w literaturze jako wzorców do oceny innych języków. Języki te cechują się też uniwersalnością zastosowań. Pominięto natomiast języki specjalistyczne, na przykład używane do programowania sterowników lub robotów przemysłowych [11]. Przykład zagadnień związanych z projektowaniem oprogramowania dla sterowników przedstawia praca [1].

Porównanie języków przeprowadzone w [11], opiera się na następujących kryteriach:

1. Szeregowalność - rozumiana jako możliwość programowego sterowania kolejnością wykonywania programu oraz jako możliwość przeprowadzania analizy szeregowalności, czyli szacowania czasów wykonywania programów.
2. Niezawodności i odporność programów - określona m.in. przez stosowanie zasady silnej typizacji, mechanizmy obsługi wyjątków, mechanizmy strukturalizacji programów.
3. Zestaw konstrukcji językowych do wyrażania i organizacji obliczeń współbieżnych.
4. Modularność - rozumiana jako możliwość rozdzielnej kompilacji i organizowania programowania w zespole programistycznym.
5. Dostęp do sprzętu - rozumiany jako możliwość tworzenia programowego interfejsu do specyficznych urządzeń zewnętrznych komputera.
6. Konserwowalność - rozumiana jako czytelność programu i możliwość wprowadzania zmian w tekście programu.
7. Efektywność implementacji.

Stopień spełnienia każdego z tych kryteriów przez dany język ocenia się w skali od 0 (brak spełnialności) do 4 (całkowite spełnienie).

Tabela jest fragmentem ocen – opracowanych w [11] – w odniesieniu do języków, które objęto przeglądem w poprzednich numerach *Informatyki*, do języków porównywanych w [6], oraz jeszcze do kilku innych, które – z powodów wyjaśnionych dalej – także zasługują na uwagę. Kolumny odpowiadają wyżej wymienionym kryterium. Brak liczby w niektórych pozycjach tabeli oznacza, że ze względu na dane kryterium trudno było przypisać językowi jednoznaczną ocenę.

Zauważmy, że najniższą ocenę mają czasowe rozszerzenia języka FORTRAN. Jest to oczywiste, gdyż są to jedne z najwcześniejszych (lata sześćdziesiąte) prób definiowania języka programowania czasu rzeczywistego, i to w oparciu język, który został utworzony w zupełnie innym celu. Niewiele wyższe oceny mają czasowe rozszerzenia, dosyć popularnego, zwłaszcza w latach siedemdziesiątych, języka PL/1. Wyraźnie wyższe notowania ma – powstały pod koniec lat siedemdziesiątych i stosowany jeszcze do niedawna, zwłaszcza we Francji – język LTR. Obecnie, częściowo z powodów politycznych, został zastąpiony przez język Ada.

Natomiast oceny najwyższe osiągnęły języki DPS (Distributed Programming System) [8], DICON (Distributed CONfiguration language) [9] i RT Euclid [3]. Wszystkie te języki - powstałe w połowie lat osiemdziesiątych – pozostają tylko językami studialnymi lub co najwyżej eksperymentalnymi. Były one projektowane od początku w oparciu o przedstawione kryteria oceny języków i dlatego wyznaczają jedynie wzorce, w oparciu o które można oceniać inne języki.

Wśród języków, które objęto przeglądem w Informatyce, najwyższe oceny mają Pearl [12] i Ada 95 [4]. Warto przypomnieć, że prace nad Pearl (Process and Experiment Automation Real-time Language) podjęto jeszcze w końcu lat sześćdziesiątych pod patronatem zachodnio-niemieckiego ministerstwa nauki i techniki. Od tego czasu język jest rozwijany i jednocześnie szeroko stosowany w przemyśle, zwłaszcza niemieckim. Ocena Ady 95 zawarta w tabeli 1 odnosi się do jej roboczej wersji z roku 1992, dlatego ocena szeregowalności języka (1-2) jest wyraźnie zaniżona. Wydaje się, że Ada 95 może w tym względzie być oceniona przynajmniej na 2-3.

Lp.	Język	Szeregowalność	Niezawodność i odporność programów	Konstrukcje językowe obliczeń współbieżnych	Modularność	Dostęp do sprzętu	Konserwowalność	Efektywność implementacji
1	RT FORTRAN	1	0-1	0-1	0-1	1-4	1-2	4
2	RT PL/1	1-2	2	2	1	3	2	
3	LTR	1	2	2	2	4	3	4
4	DPS, DICON	4	4	4	4	4	4	4
5	RT Euclid	4	4	4	4	4	4	3
6	Occam, Occam 2	0	2	4	4		3	
7	Chill	1	4	4	4	3	4	
8	Pearl	2-3	3	2	4	4	1	3
9	Distributed Pearl	2-3	3	2	4	4	4	
10	Ada 83	1	4	4	4	4	3	2-4
11	Ada 95*)	1-2	4	4	4	4	3	3-4

*) Ocena odnosi się do roboczej wersji Ady z 1992 roku.

Przedstawiona tabela nie wyczerpuje możliwości porównań języków. Poza tabelą pozostają jeszcze inne aspekty dotyczące języków programowania. Szczególnie ważne aspekty, które należy brać pod uwagę w sytuacjach praktycznych, na przykład przy wyborze języka do realizacji projektów oprogramowania systemów czasu rzeczywistego, to:

- obszar zastosowań,
- środowiska programistyczne i ich rozpowszechnienie,
- związek języka z metodami i narzędziami projektowania systemów czasu rzeczywistego,
- standaryzacja i perspektywy rozwoju języka,
- dostępność i rozpowszechnienie języka.

Aspekty te są często ze sobą powiązane tak, że nie można rozpatrywać ich oddzielnie. Biorąc to pod uwagę, dla języków objętych przeglądem, można przedstawić następujące uwagi.

RT Euclid - jak wspomniano - jest językiem studialnym i to eliminuje go z zastosowań praktycznych. Może pełnić co najwyżej rolę wzorca, do którego warto odwoływać się przy ocenie innych języków.

Obszar zastosowań języka Occam nie jest zarysowany w jakiś szczególny sposób, ale największa liczba zastosowań wiąże się z przetwarzaniem sygnałów lub przetwarzaniem graficznym. Zasadnicze ograniczenie w upowszechnieniu języka wiąże się z tym, że Occam jest przeznaczony do programowania transputerów, specjalizowanych mikroprocesorów produkowanych przez brytyjską firmę Inmos Ltd. Transputery są montowane na kartach, które można instalować na komputerach typu IBM PC. Dodatkowo na komputerach PC jest instalowane profesjonalne środowisko programistyczno-uruchomieniowe TDS (*Transputer Development System*) zawierające, m.in., edytor, kompilator, debuger-analizator, bibliotekę funkcji numerycznych i procedur wejścia/wyjścia.

Zakres zastosowań języka Chill jest silnie związany z telekomunikacją. Prace nad jego rozwojem były prowadzone pod patronatem CCITT (obecnie ITU - *International Telecommunication Union*), a głównymi użytkownikami są duże firmy telekomunikacyjne tworzące oprogramowanie dla central telefonicznych i abonenckich w sieciach publicznych. Miarą rozpowszechnienia języka jest liczba używających go programistów, szacowana na około 15 tysięcy. Jest to interesujące tym bardziej, że właściwie jest oferowane tylko jedno komercyjne środowisko programistyczne CHIPSY, przez norweską firmę Kvatro [13]. Istotne znaczenie dla przyszłości języka ma bliskość koncepcyjna jego mechanizmów z językiem specyfikacji SDL (*System Description Language*), który jest również standardem CCITT. SDL jest językiem specyfikacji wykonywalnych i dzięki temu możliwa staje się automatyzacja przejścia od fazy specyfikacji w SDL do implementacji w języku Chill.

Pearl zasługuje na szczególną uwagę zarówno ze względu na swoją historię jak i swoją wartość. Od momentu powstania jest szeroko stosowany w badaniach, przemyśle, a także w obronności. Powstały imlementacje języka praktycznie na wszystkie dostępne platformy sprzętowe. Oferowane są liczne profesjonalne środowiska programistyczne, a niektóre z nich są dostępne w wersjach bezpłatnych. Język jest nadal rozwijany o czym świadczą kolejne normy DIN, ostatnia z norm odnosi się do wersji Distributed Pearl z 1990 roku. Można uznać, że Pearl zyskał rangę niemieckiego, narodowego języka programowania systemów czasu rzeczywistego.

Ada powstała na początku lat osiemdziesiątych w wyniku inicjatywy Departamentu Obrony USA i wiązała się z zamierzeniem pełnej unifikacji oprogramowania w zastosowaniach militarnych oraz w związanym przemyśle. Już ta przesłanka wskazuje na szczególną rolę Ady i perspektywy jej rozwoju. Dodatkowym impulsem dla rozpowszechnienia Ady stały się prace nad nową wersją – Adą 95, prowadzone w uzgodnieniu z Międzynarodową Organizacją Normalizacyjną (ISO). Obecnie pojawiają się na rynku pierwsze pełne implementacje języka Ada 95 wraz ze zintegrowanymi środowiskami programistycznymi na większość współcześnie używanych platform sprzętowych. Są też oferowane implementacje bezpłatne, m.in., na komputery typu IBM PC z systemami operacyjnymi DOS, Windows 95, Windows NT. Dodatkowym atutem dla dalszego rozpowszechnienia Ady jest powstawanie metod projektowania oprogramowania ukierunkowanych na wykorzystanie Ady. Przykładem takiej metody jest HTR-HOOD™ [1], stanowiąca rozwinięcie wcześniejszej metody HOOD, opracowanej dla projektowania oprogramowania w Adzie 83.

Z przytoczonych uwag nie można wnioskować, że przyszłość będzie zdominowana przez jeden z omawianych języków. Realną natomiast wydaje się perspektywa współistnienia języków Chill, Pearl czy Ady, przynajmniej przez okres następnych kilku lat. W niektórych dziedzinach zastosowań - na przykład Chill w telekomunikacji, czy Ada w zastosowaniach militarnych USA – będą wyraźnie przeważać nad innymi językami. Trudno przewidzieć czy w okresie tym nie pojawią się nowe idee i związane z nimi języki. Można, na przykład, spodziewać się, że rozwijane obecnie języki programowania obiektowego, takie jak C++, Eiffel czy Smalltalk, będą oferować nowe wersje ukierunkowane na programowanie systemów czasu rzeczywistego.

Literatura

[1] Burns A., Wellings A., *HRT-HOOD™: A Structured Method for Hard Real-Time Ada Systems*, Elsevier, 1995

[2] Buczyński T., Loska J., Ober J., *Zastosowanie programu BetterState do projektowania systemów czasu rzeczywistego*, Materiały III Konferencji Systemów Czasu Rzeczywistego, Szklarska Poręba, 10-13 września 1996, Oficyna Wydawnicza Politechniki Wrocławskiej, s. 248-256, 1996

[3] Complak W., Nawrocki J.R., *Real-Time Euclid: Język programowania systemów silnie uwarunkowanych czasowo*, Informatyka nr 10, 1996

[4] Fryźlewicz Z., Huzar Z., *Programowanie systemów czasu rzeczywistego w Adzie 95*, Informatyka nr 3, 1997

[5] Glass R.L. (ed.), *Real-Time Software*, Prentice-Hall, Inc., 1983.

[6] Halang W., Sacha K.M.: *Real-Time Systems. Implementation of Industrial Computerised Process Automation*, World Scientific, 1992

[7] Huzar Z., *Języki programowania czasu rzeczywistego*, Informatyka nr 10, 1996

[8] Lee I., *Programming System for Distributed Real-Time Applications*, Proc. of the IEEE1984 Real-Time Symposium, 1984, pp. 18-27

[9] Lee I., Gehlot V., *Language Constructs for Distributed Real-Time Programming*, Proc. of the IEEE1985 Real-Time Symposium, 1985, pp. 57-66

[10] Mikanik W., *Charakterystyka języka occam*, Informatyka nr 11, 1996

[11] Stoyenko A. D., *The Evolution and State-of-theArt of Real-Time Languages*, The Journal of Systems Software, pp. 61-84, 18/1992

[12] Szmuc T., Nowak W., *Wprowadzenie do języka PEARL 90*, Informatyka nr 1 i nr 2, 1997

[13] Średniawa M., Brzeziński K., *Język programowania CHILL*, Informatyka nr 12, 1996.

prof. Zbigniew Huzar, Wydziałowy Zakład Informatyki, Wydział Informatyki i Zarządzania, Politechnika Wrocławska

OGÓLNOPOLSKA KONFERENCJA

SYSTEMY INFORMATYCZNE W PRZEMYŚLE

II TARGI MRP

5-6 czerwca 1997 r.,
Warszawa, PKiN, VI p.

Do udziału w konferencji zapraszamy przedstawicieli ścisłego kierownictwa oraz głównych informatyków jednostek przemysłowych

PROGRAM RAMOWY

5.VI.1997 r. (czwartek)

10.00 Otwarcie, wystąpienie Ministra Gospodarki - Wiesława Kaczmarka n.t. polityki gospodarczej państwa.
10.30 "BPR" Business Process Reengineering - prof. Stig Arne Mattson (Uniwersytet w Upsali - Szwecja).
12.30 Przerwa.
13.00 Prezentacje firm - wystawców.
14.30 Obiad.
16.00 Konfigurator produktów - Marek Greniewski.
16.45 Analiza krajowego rynku odbiorców systemów klasy MRP - Andrzej Dyżewski.
17.30 Przerwa.
17.50 Nowe technologie i zarządzanie w przemyśle - Prezes Tadeusz Soroka - Agencja Techniki i Technologii.
18.30 Konsultacje w stoiskach firm - wystawców.
20.00 Coctail.

6.VI.1997 r. (piątek)

9.00-11.00 Konsultacje w stoiskach firm - wystawców.
10.00 Metodologia MRP II - Aleksander Popończyk.
11.00 "BPR" Business Process Reengineering na przykładzie polskiego przedsiębiorstwa.
11.40 Przerwa.
12.00 Prezentacje firm - wystawców.
14.00 Obiad.
15.00 Dyskusja panelowa nt.: "Uwarunkowania wdrażania MRPII w Polsce" - moderator dyskusji Tadeusz Soroka - Prezes Agencji Techniki i Technologii.
16.30-18.00 Konsultacje w stoiskach firm - wystawców.

Pod patronatem:
Ministra Gospodarki
Agencji Techniki i Technologii

Patronat medialny:
COMPUTER WORLD
INFORMATYKA
RZECZPOSPOLITA

Organizacja:
PTI - Oddz. Górnośląski
Centrum Promocji Informatyki

Informacje organizacyjne:

1. Termin organizacji konferencji 5-6.VI.1997 r.
2. Miejsce organizacji PKiN w Warszawie, Sala Koncertowa VI p.
3. Koszt udziału 1 osoby obejmuje: 490 zł + 22% VAT
   - udział w konferencji i targach,
   - 2 lunchy i przerwy na kawę,
   - materiały,
   - udział w koktajlu
4. Zgłoszenia udziału wg. załączonego wzoru prosimy przesyłać pocztą, faxem lub e-mailem:

   PTI - Oddział Górnośląski
   40-012 Katowice, ul. Św. Jana 10
   tel. fax: (048-32) 1539678
   fax: (048-32) 7346140
   e-mail: ptiog@netpol.pl

5. Termin nadsyłania zgłoszeń: do 28.V.1997 r.
   Uwaga! Ilość miejsc ograniczona, decyduje kolejność zgłoszeń.
6. Rezerwacja miejsc hotelowych:

Hotel	pokój jednoosobowy (ze śniadaniem)	pokój dwuosobowy (ze śniadaniem)
GRAND	220,00 + 22% VAT /z łazienką/	----------
METROPOL	180,00 + 22% VAT /z łazienką/	245,00 + 22% VAT /z łazienką/
POLONIA	90,00 + 22% VAT /z umywalką/	225,00 + 22% VAT /z łazienką/

Zamówienia dotyczące rezerwacji miejsc hotelowych należy przekazać najpóźniej do dnia: 23.05.97r.

Firmy teleinformatyczne zainteresowane udziałem w ekspozycji prosimy o kontakt:
p. Agnieszka Markowska (022) 621.76.26, 693.59.46,
e-mail: agnieszka@cpi.com.pl

Centrum Promocji Informatyki Sp. z o.o.,
00-503 Warszawa 15, ul. Żurawia 4a, skr. p. 44,
tel. (048-22) 6935946, 6217626, 6935922
fax: (048-22) 6935958, 6935949

Dokładna nazwa i adres firmy

NIP

INFORMATYKA

Zgłoszenie udziału w konferencji: SYSTEMY INFORMATYCZNE W PRZEMYŚLE - II TARGI MRP

Zgłaszamy udział następujących przedstawicieli n/Firmy:

imię i nazwisko	stanowisko	tel.	fax:	e-mail:

Rezerwacja hoteli (prosimy wpisać nazwę hotelu, czas rezerwacji oraz rodzaj pokoju): .....

I. Zobowiązujemy się do uregulowania należności z tytułu zgłoszenia udziału w/w osób w konferencji po otrzymaniu faktury, w terminie 7 dni od daty fakt upoważniamy PTI do wystawienia faktury VAT bez naszego podpisu.

II. Zobowiązujemy się do uregulowania należności z tytułu rezerwacji miejsc hotelowych po otrzymaniu faktury. Upoważniamy Agencję Inter-Media do wystawie faktury VAT bez naszego podpisu.

POLSKIE TOWARZYSTWO INFORMATYCZNE

BIULETYN

NUMER 5 (134) ROK XVI MAJ 1997

PTI MA 10000_2 LAT !

Szanowni Państwo,
22 maja bieżącego roku mija 10000_2 lat od chwili założenia naszego Towarzystwa. W tym czasie kierowało nim 100_2 Prezesów (aktualnego włączając), a dochowaliśmy się ponad 1000000000000_2 członków. Z okazji tego jubileuszu życzmy sobie jak najszybszego podwojenia naszych szeregów, wielu udanych przedsięwzięć i doczekania w dobrej zawodowej kondycji binarnej „milionowej" rocznicy.

Sprawozdanie z posiedzenia Zarządu Głównego PTI w dniu 18 kwietnia 1997

W posiedzeniu uczestniczyło 9 członków Zarządu oraz zaproszeni goście

- Prezes ZG, Kol. Zdzisław Szyjewski poinformował o spotkaniu konsultacyjnym, jakie odbyło Prezydium ZG z Polską Izbą Informatyki i Telekomunikacji, które poświęcone było m.in. powołaniu instytucji rzeczoznawców PTI.
- Prezes zdał krótkie sprawozdanie ze spotkania polskiego środowiska informatycznego z goszczącym w Polsce Martinem Bangemannem, Wysokim Komisarzem Unii Europejskiej, na które zaproszone zostały 3 osoby z PTI. Przedstawiono proces wprowadzania ECDL w Polsce, co zyskało pozytywny oddźwięk i uwzględnione zostanie w Europejskim Raporcie o Globalnym Społeczeństwie Informacji.
- Kol. Zygmunt Mazur przedstawił zasady XIV Ogólnopolskiego Konkursu PTI na najlepsze prace magisterskie z informatyki. Zarząd zatwierdził skład Jury oraz wysokość nagród na poziomie ubiegłorocznym.
- Kol. Jerzy Nowak przedstawił wyniki wstępnych ustaleń dotyczących powołania Izby Rzeczoznawców PTI. Proponuje się, by PTI było odpowiedzialne za akwizycję zleceń i ich finansowe rozliczenie, natomiast odpowiedzialność prawna spoczywałaby na wykonawcy ekspertyzy. Ustalono, że projekt funkcjonowania Izby przedstawiony zostanie do dnia 15 maja 1997.
- Kol. Marek Miłosz poinformował o postępach we wprowadzaniu ECDL w Polsce. Dotychczas przeprowadzono ok. 30 egzaminów. Pierwsze egzaminy odbyły się we Wrocławiu. W Krakowie MEN zorganizowało spotkanie koordynatorów nauczania informatyki poświęcone ECDL, w Warszawie egzaminami zainteresowane są szkoły podstawowe i średnie, a także Instytut Maszyn Matematycznych. W Opolu Zespół Szkół zawodowych zaangażowany w ustawiczne kształcenie dorosłych wraz z PTI oraz ośrodkiem w Danii uzyskał fundusze PHARE na realizację ECDL. Rolą PTI w tym przedsięwzięciu jest obsługa administracyjna i przygotowanie programów nauczania. Kol. M. Miłosz wyraził potrzebę większej liczby egzaminatorów. 18 kwietnia w siedzibie InfoVide przeszkolono kolejnych 12 osób, zgłoszono propozycję przeprowadzenia szkolenia w Poznaniu.
- Zaproponowano utworzenie Komisji Akredytacji Programów Nauczania. Jednogłośnie przyjęto kandydaturę Kol. Jurka Nawrockiego na jej Przewodniczącego i powierzono mu zadanie sformowania składu Komisji.
- Zatwierdzono bilans PTI za rok 1996 i zapoznano się z projektem budżetu na rok 1997.
- Przewodnicząca Głównej Komisji Rewizyjnej PTI, Kol. Teresa Tomaszewska przedstawiła sprawozdanie z posiedzenia z dnia 17.04.97, w którym zawarte były wnioski o oszczędne gospodarowanie dysponowanymi środkami oraz wniosek do Sekretarza o konsultowanie zmian w składzie Biura ZG z Zarządem Głównym.
- W związku ze służbowym wyjazdem Sekretarza ZG, Kol. Leszka Bogusławskiego i spowodowaną tym jego nieobecnością, przewidzianą w porządku obrad informację o działalności Biura przełożono na następne spotkanie
- Przyjęto 10 nowych członków PTI. Trzy aplikacje odrzucono ze względów formalnych.

XIV OGÓLNOPOLSKI KONKURS NA NAJLEPSZE PRACE MAGISTERSKIE Z INFORMATYKI

W konkursie mogą brać udział wszyscy dyplomanci studiujący na wyższych uczelniach w Polsce (również obywatele innych krajów), których prace dyplomowe dotyczą informatyki i którzy złożyli i obronili te prace w terminie określonym przez regulamin studiów odnośnej uczelni.

Zgłoszenia pracy na konkurs dokonuje autor (autorzy) pracy, przysyłając na adres organizatora konkursu:

POLSKIE TOWARZYSTWO INFORMATYCZNE
ul. Pawła Włodkowica 8 p. 18,
50-072 Wrocław
tel. 44 52 79, fax 324 95 70

w terminie do 5 października 1997 następujące dokumenty:

a) pracę dyplomową wraz z załącznikami i niezbędnymi do dokonania oceny (dyskietka z oprogramowaniem, zaświadczenie o wdrożeniu, kopia publikacji itp.)
b) wypełnioną przez autora (autorów) ankietę (formularze ankiet wysłano do dziekanatów i instytutów oraz są także dostępne u organizatorów konkursu)

c) zaświadczenie z uczelni, że zgłoszona na konkurs praca została obroniona jako praca magisterska w roku ogłoszenia konkursu, tj. w okresie od 1. października 1996 r. do 30 września 1997 r.

WYNIKI KONKURSU BĘDĄ OGŁOSZONE NIE PÓŹNIEJ NIŻ DO KOŃCA 1997 ROKU

NAGRODY

I	nagroda	1200 zł
II	nagroda	900 zł
III	nagroda	700 zł
trzy wyróżnienia po		500 zł

Komisja konkursowa może nie przyznać dowolnej z nagród, bądź podzielić jedną nagrodę pomiędzy kilka prac. Po zakończeniu konkursu przedłożone prace magisterskie zostaną zwrócone autorom.

Zapraszamy do wzięcia udziału w konkursie
DOLNOŚLĄSKI ODDZIAŁ PTI

Organizatorzy składają podziękowanie dr. Witoldowi Staniszkisowi - Prezesowi firmy RODAN - SYSTEM za ufundowanie I nagrody w wysokości 1200 zł w XIII edycji Konkursu

Ożywienie działalności Koła PTI Gdańsku

19 kwietnia 1997 r. odbyło się zebranie naszego Koła. Z radością odnotowujemy ten fakt, ponieważ nasze Koło od pewnego czasu formalnie istniało, ale działalność jego była znikoma.

Zebraniu przewodniczył Prezes Koła – kol. Franciszek Gryka. Uczestniczyli w nim nowo przyjęci członkowie, którzy deklarowali aktywny udział w pracach PTI. Na spotkanie przybył również Prezes ZG PTI – kol. Zdzisław Szyjewski, który omówił statutowe cele działalności PTI oraz bieżące jego prace. Wobec dużego zainteresowania Komputerowym Prawem Jazdy – Prezes ZG PTI omówił procedurę oraz tryb wprowadzania ECDL i zachęcił członków Koła do włączenia się w tę inicjatywę. Dyr. Ośrodka Informatyki Urzędu Wojewódzkiego w Gdańsku, kol. Jolanta Sala złożyła konkretną propozycję współpracy Koła z ZG w tym zakresie. Ustaliliśmy, że podstawowymi kierunkami działania w najbliższej przyszłości będą:

- pozyskiwanie nowych członków Koła
- szkolenia i seminaria tematyczne
- promocja, szkolenia i egzaminy ECDL

W trakcie dyskusji zebrani wygenerowali kilkanaście tematów, które będą prezentowane w ramach kolejnych spotkań.

Na nastepne dwa spotkania wiodącym tematem będzie: „Europa i Społeczeństwo Globalnej Informacji – Raport Bangemanna".

Alicja Myszor

TO NIE SEN - TO JAVA

II Letnia Szkoła Programowania Wizualnego Poznań '97

TEMATYKA:

Druga edycja Letniej Szkoły Programowania Wizualnego organizowanej przez Oddział Wielkopolski PTI poświęcona będzie metodom tworzenia oprogramowania za pomocą języka Java. Podczas wykładów i ćwiczeń laboratoryjnych wykładowcy zaproszeni z Centre for Teaching Computing, Dublin City University oraz pracownicy Instytutu Informatyki Politechniki Poznańskiej przedstawią następujące zagadnienia:

- Wprowadzenie (co to jest Java, do czego można ją wykorzystać, środowisko pracy),
- Programowanie w języku Java (typy danych, zmienne, wyrażenia, operacje na tablicach i ciągach znaków, instrukcje sterujące, obiekty i klasy, interfejsy, pakiety, wyjątki, wątki),
- Przegląd standardowych pakietów klas,
- Programowanie appletów (dziedziczenie z klasy bazowej, fazy życia appleta, graficzny interfejs użytkownika, wyświetlanie obrazów, odtwarzanie dźwięku, animacja),
- Operacje na plikach, programowanie sieciowe, łączenie języka Java z językiem C.
- Prowadzone równolegle laboratoria obejmować będą:
- Kompilacja i uruchamianie programów prezentowanych na wykładach,
- Programowanie własnych aplikacji i appletów,

Zajęcia prowadzone będą **w języku angielskim** codziennie w godzinach **od 9.00 do 17.00.** Od uczestników oczekuje się znajomości podstaw programowania obiektowego.

TERMIN:

9 - 11. 09. 1996

MIEJSCE:

Instytut Informatyki Politechniki Poznańskiej
ul. Piotrowo 3A
60-965 Poznań

OPŁATA:

Opłata za udział w Letniej Szkole Programowania Wizualnego wynosi **350** zł (dla członków PTI - **300** zł). W ramach tej sumy uczestnicy otrzymują komplet materiałów, a także przez trzy dni obiad oraz poczęstunki podczas przerw. Dla uczestników spoza Poznania organizatorzy mogą zarezerwować hotel.

ZGŁOSZENIA:

Zgłoszenia pisemne można nadsyłać do dnia **10.08.1996**. Ze względu na ograniczoną liczbę stanowisk laboratoryjnych o przyjęciu na Letnią Szkołę Programowania Wizualnego decydować będzie kolejność zgłoszeń. Po przyjęciu zgłoszenia organizatorzy prześlą szczegółowy program wraz z informacją o sposobie wniesienia opłaty.

KOMITET ORGANIZACYJNY:

Jan Kniat, Ewa Łukasik, Mikołaj Sobczak, Sergiusz Strykowski, Adam Wojciechowski.

Telefony: (61) 782-373, 782-372, 782-364, 782-369
Fax: (61) 771-525
E-mail: java@cs.put.poznan.pl

Zapraszamy do odwiedzania strony WWW poświęconej Letniej Szkole Programowania Wizualnego http://www.cs.put.poznan.pl/~java

I REGIONALNA KONFERENCJA INFORMATYCZNA

Kazimierz Dolny n/Wisłą 21-25.04.1997r

Informatyczne środowisko lubelskie nie posiadało do tej pory forum do wymiany poglądów nt. problemów nauczania informatyki i kierunków badań naukowych, związanych z informatyką lub realizowanych przy jej pomocy. Pewną namiastką takiego forum były organizowane od 1989 roku Warsztaty Mikrokomputerowe Politechniki Lubelskiej. Warsztaty te w zeszłym roku zbiegły się w czasie i miejscu z dorocznymi Konferencjami Metodycznymi Kuratorium Oświaty w Lublinie. Tak zrodziła się idea Regionalnej Konferencji Informatycznej.

Pierwsza Regionalna Konferencja Informatyczna odbyła się w Kazimierzu nad Wisłą w dniach 21-25.04.1997 r. W jej skład wchodziły następujące imprezy:

- **Lubelskie Akademickie Forum Informatyczne**
- **VI Warsztaty Mikrokomputerowe Politechniki Lubelskiej**
- **Konferencje Metodyczne Kuratorium Oświaty w Lublinie**
- **Spotkanie Polskiego Klubu Użytkowników Systemu Progress**
- **Spotkanie Koła Polskiego Towarzystwa Informatycznego w Lublinie**

Każda z imprez miała swoje grono odbiorców i uczestników. W ich trakcie doszło do wymiany informacji, spotkań formalnych i nie-, zacieśnienia stosunków pomiędzy informatykami z wszystkich uczelni miasta Lublina, a także nauczycielami podstaw informatyki szkół średnich i podstawowych. W Kazimierzu gościli także programiści, tworzący oprogramowanie w systemie Progress - jednym z nowocześniejszych systemów do tworzenia aplikacji wspomagających zarządzanie przedsiębiorstwem. Spotkanie Polskiego Klubu Użytkowników Systemu Progress tym razem było poświęcone Internetowi.

Jednym słowem przez Kazimierz w ostatnim tygodniu kwietnia przewinęło się ponad 250 osób związanych z informatyką w naszym regionie.

Lubelskie Akademickie Forum Informatyczne jest spotkaniem dydaktyków i naukowców uczelni lubelskich. Jest ono poświęcone dwóm wiodącym tematom: nauczanie informatyki w wyższych uczelniach Lublina i informatyka w badaniach naukowych. Udział w Forum zapowiedzieli przedstawiciele MENu.

Szóste już z kolei Warsztaty Mikrokomputerowe Politechniki Lubelskiej są poświęcone dydaktyce informatyki w Politechnice Lubelskiej, eksperymentalnemu programowi nauczania infromatyki w szkole podstawowej nr 13 w Lublinie, matematyce w informatyce i wybranym zagadnieniom informatyki współczesnej. Warsztaty odbywają się tradycyjnie w Domu Pracy Twórczej Politechniki Lubelskiej.

Konferencje Metodyczne Kuratorium Oświaty w Lublinie gromadzą nauczycieli informatyki szkół średnich i podstawowych. Są one poświęcone koordynacji programów nauczania podstaw informatyki. Odbywają się w murach Liceum Ogólnokształcącego.

Spotkanie Polskiego Klubu Użytkowników Systemu Progress poświęcone jest tematyce biznesowych zastosowań Internetu. Zgromadzi ono ponad 40 uczestników z różnych miast Polski: Warszawy, Gdańska, Zielonej Góry, Łodzi, Katowic itd. Przedstawiciele firmy CSBI S.A. zaprezentowali sztandarowy produkt Progressa do pracy w Internecie - WebSpeed.

W trakcie spotkania Lubelskiego Koła Polskiego Towarzystwa Informatycznego odbyło się szkolenie egzaminatorów Europejskiego Komputerowego Prawa Jazdy (EKPJ).

Organizatorami Pierwszej Regionalnej Konferencji Informatycznej są:

- Polskie Towarzystwo Informatyczne - Koło w Lublinie (PTI)
- Zakład Systemów Informatycznych Katedry Zarządzania Politechniki Lubelskiej (PL)
- Katedra Informatyki Politechniki Lubelskiej (PL)
- Kuratorium Oświaty w Lublinie (KO)
- Polski Klub Użytkowników Systemu Progress (PKUSP)

Jak przystało na informatyków Konferencja ma swój serwis w Internecie. Szczęśliwi posiadacze dostępu do sieci mogą znaleźć szczegółowe informacje pod adresem:
http://pluton.pol.lublin.pl/kaz97.htm

Redaktor: EWA ŁUKASIK
e-mail: lukasik@poznlv.put.poznan.pl, tel. (61) 782 373,
Instytut Informatyki, Politechnika Poznańska,
ul. Piotrowo 3a, 60-965 Poznań

Polskie Towarzystwo Informatyczne, Zarząd Główny
tel. (22) 624 60 61 w.328, tel./fax (22) 652 32 59
URL: http://www.pol.pl/pti
ul. Żelazna 87, 00-879 Warszawa

I N T E R N E T KONCESJA 29/96/J

File Edit Window Help

BACK NEXT STOP HOME LOOK BEST CODE PRINT HELP

Site: http://heros.juni.com/
http://heros.juni.gliwice.pl./

1987 – 1997

JUNISOFTEX OFERUJE:

- Nowoczesne obiektowe aplikacje
- Zarządzanie procesem projektowym wsparte narzędziami CASE (w tym Select OMT)
- Analizy dziedzinowe oparte o metodę łączącą nowoczesność z 20 letnim doświadczeniem wdrożeniowym
- Sprawne i szybkie wdrożenia zgodne z obowiązującymi przepisami
- Projektowanie i wykonawstwo sieci komputerowych
- Typowe, publiczne usługi internetowe dla osób prawnych i fizycznych
- Technologie internetu do restrukturyzacji procesów organizacyjnych
- Technologie internetu w systemach wspomagających proces zarządzania
- Integracje różnych rodzajów sieci
- dostawy sprzętu komputerowego renomowanych firm
- dostawy sprzętu peryferyjnego

Perfect-Ekspert '97

System Perfect-Ekspert'97 jest zbiorem aplikacji integrujących procesy gospodarcze w całym przedsiębiorstwie.

Strategia zarządzania i systemy informowania kierownictwa
Sprawozdawczość Obserwacja Dociekanie Wnioskowanie Komentowanie Symulacje i Prognozy

System finansowo-kosztowy
Banki Kasa Sprzedaż Zakup Koszty proste Rozliczanie kosztów Sprawozdawczość

System ewidencji majątku
Ewidencja Inwentaryzacja Koszty Plan amortyzacji Przeszacowania Sprawozdawczość

Systemy obsługi klienta, obrót towarowy, sprzedaż
Zamówienia Ewidencja Inwentaryzacja Zakup towarów Kompletacja Ewidencja opakowań zwrotnych Sprzedaż Analizy Sprawozdawczość

Systemy gospodarki materiałowej i zaopatrzenia
Zakupy Ewidencja Inwentaryzacja Dyspozycja Rezerwacja Koszty Analiza Sprawozdawczość

System kadrowo-płacowy oraz dyscypliny
Dokumentacja i ewidencja Sytemy wynagrodzeń Czas pracy i rejestratory czasu pracy Kwalifikacje i szkolenia Emisja dokumentacji ZUS i zasiłki Sprawozdawczość

System przygotowania produkcji
Dokumentacja technologiczna Narzędzia i urządzenia pomiarowe Kalkulacje i ofertowanie Emisja dokumentacji warsztatowej Plan zakupów Plany warsztatowe Produkcja w toku Rozliczenie produkcji

Systemy pomiarowe i stanowiska pomiarowe
Statystyczne oceny jakości Dokumentacja jakości Świadectwa jakości i atesty Baza pomiarowa

Przedsiębiorstwo Produkcyjno–Usługowe Innowacji Technicznych
JUNISOFTEX® Sp. z o.o.
ul. Kościuszki 1c, 44-100 GLIWICE
tel.: (0-32) 130 69 31 do 33
fax: (0-32) 130 69 34
E-mail: DYREKTOR@JUNIGLIWICE.PL
E-mail: PREZES@JUNIGLIWICE.PL
Nr Polpak: 13211054
NIP: 631-020-03-11
Regon: P-008012971-94927177-59-3-671-27177
Konto: PKO I/O Gliwice 27661-225298-136
Dyrektor Naczelny – Krzysztof Niewiadomski
Prezes – Bronisław Jureczko
Kapitał zakładowy 1.150.000.000 PLZ (115.000 PLN)

Firma JUNISOFTEX obecna jest na rynku od 10 lat. W tym czasie wdrożono kilkaset dużych aplikacji. Naszej firmie zaufało jak do tej pory około 100 przedsiębiorstw produkcyjnych w tym 6 spółek giełdowych, 10 przedsiębiorstw NFI, kilka przedsiębiorstw przemysłu hutniczego i wiele innych.

Lista aktualnych użytkowników „Perfect-Ekspert"

- Elektrownia RYBNIK w Wielopolu
  jedyna w Polsce elektrownia rozliczająca koszty rzeczywiste remontów na 3 tys. obiektów
- ELEKTROBUDOWA S.A. w Katowicach
- POLIFARB S.A. w Cieszynie
- WEDEL w Warszawie
- P.P.O.W. AGROS-PIŃCZÓW w Pińczowie
- Zakłady Odzieżowe BYTOM S.A. w Bytomiu
- Maszyny Elektryczne CELMA w Cieszynie
- Fabryka Amortyzatorów KROSNO S.A. w Krośnie
- Instytut Spawalnictwa w Gliwicach
- GEOFIZYKA w Toruniu
- POLSPORT w Bielsku-Białej
- POLAM-KOSTUCHNA w Katowicach
- Walcownia Metali DZIEDZICE S.A. w Czechowicach-Dziedzicach
- ELEKTRONARZĘDZIA CELMA S.A. w Cieszynie
- Zakłady Tworzyw Sztucznych KRYWAŁD ERG w Knurowie
- Fabryka Aparatów Elektrycznych APENA S.A. w Bielsku-Białej
- MOSTOSTAL Słupca
- Fabryka Osi Napędowych w Radomsku
- Huta Szkła Okiennego KUNICE S.A. w Żarach
- Huta ŁAZISKA w Łaziskach Górnych
- ELEKTROMONTAŻ 3 S.A. w Katowicach
- Fabryka Palenisk Mechanicznych w Mikołowie
- DRAGMOR sp. z o.o. w Szczecinie
- Dąbrowskie Zakłady Naprawcze Przemysłu Węglowego w Dąbrowie Górniczej
- Centrum Naukowo Techniczne Kolejnictwa w Warszawie
- Zakłady Rowerowe ROMET Kowalewo
- SUWARY S.A. w Pabianicach
- Zakłady Naprawcze Przemysłu Węglowego w Rybniku
- Rejonowe Przedsiębiorstwo Wodociągów i Kanalizacji w Gliwicach
- Zakłady Tłuszczowe BIELMAR S.A. w Bielsku-Białej
- GZUT w Gliwicach
- MEFTA w Mikołowie
- KRONOPOL w Żarach
- Huta BANKOWA
- Walcownia Metali ŁABĘDY S.A. w Gliwicach
- Huta 1-go MAJA w Gliwicach
- ELEKTROMETAL Zakład Telemechaniki Górniczej w Cieszynie
- Zakłady Metalurgiczne TRZEBINIA w Trzebini
- Huta SILESIA w Rybniku
- Fabryka Automatyki Chłodniczej S.A. w Cieszynie
- Zakłady Górniczo-Hutnicze BOLESŁAW w Bukownie
- MIFAMA w Mikołowie

SOFTBANK® S.A.

SOFTBANK® S.A.

BANKOWOŚĆ I SYSTEMY BANKOWE

02-146 Warszawa, ul. 17-go Stycznia 72 a
tel. 846 54 71, fax 846 14 57